

An EMAP
Publication

£2.25

The 1985 sinclair user annual

ALL THAT MADE THE NEWS IN THE SINCLAIR YEAR

**25 PAGES OF NEW
PROGRAM LISTINGS**

**THE FULL
QL STORY**

**BUYER'S GUIDE
TO JOYSTICKS
PRINTERS AND
KEYBOARDS**

**TOP SOFTWARE
AUTHOR
INTERVIEWED**

**HITS AND MISSES
IN SOFTWARE IN 1984**

**ALL YOU NEED TO KNOW
ABOUT SINCLAIR
COMPUTING**



dktronics

The Spectrum Connection



Need we say more!

DK Tronic Ltd., Saffron Walden, Essex CB11 3AQ. Tel: (0799) 26350 Telex: 81653 ATTN DK

**sinclair
user**

Editor
Bill Scolding
Deputy editor
John Gilbert
Consultant editor
Mike Johnston
Staff writer
Chris Bourne
Illustrator/designer
Craig Kennedy
Advertisement manager
Martin Derx
Advertisement secretary
Claudia Viertel
Production assistant
James McClure
Editorial assistant
Colette McDermott
Subscriptions manager
Carl Dunne
Assistant publisher
Neil Wood
Publisher
Gerry Murray
Sinclair User is published monthly
by EMAP Business & Computer
Publications

ABC

96,271
Jan-June 1984

MEMBER OF THE
AUDIT BUREAU OF
CIRCULATIONS

Telephone
Editorial and advertising departments
01-430 1200
If you would like to contribute to
Sinclair User please send
programs or articles to:
Sinclair User,
EMAP Business & Computer Publications,
67 Clerkenwell Road,
London EC1R 5BH
Original programs should be on cassette and
articles should be typed. We cannot
undertake to return them unless a
stamped-addressed envelope is
included.
We pay £10 for the copyright of each
program printed and £50
per 1,000 words for each article used.
All subscription enquiries to
Magazine Services,
EMAP Business & Computer Publications,
Priory Court,
30-32 Farringdon Lane,
London EC1R 3AU
Telephone 01-251 6222
© Copyright 1984
Sinclair User
ISSN No 0262-5458
Printed and typeset by
Cradley Print PLC,
Warley,
West Midlands
Distributed by
EMAP National Publications Ltd

CONTENTS

CHAPTER 1. News	5
<i>Chip shortages ★ Interface 1 and microdrives ★ Disappearing computers ★ Record sales of Spectrums ★ ZX Printer discontinued ★ Interface 2 disappointment ★ New software ★ Maturity of market ★ Demise of software companies ★ Pocket money games ★ Software piracy ★ QL launch ★ Delivery delays ★ Sir Clive and the critics ★ QL goes retail ★ Sinclair abandons mail order ★ The Fifth Generation ★ MSX</i>	
CHAPTER 2. Hardware World	13
<i>Joysticks round-up ★ Keyboards round-up ★ Printers round-up ★ Best of the rest</i>	
CHAPTER 3. The QL Story	
<i>A new computer is born ★ Development of an idea ★ A compromise chip ★ Networking ★ Using microdrives ★ The keyboard ★ SuperBasic ★ Psion software ★ 28 days ★ Bugs ★ The dongle ★ Delivering the goods ★ The future</i>	
CHAPTER 4. Hit Squad	39
<i>Mike Singleton, authors of Lords of Midnight and Doomdark's Revenge</i>	
CHAPTER 5. Software Scene	45
<i>The 1984 Spectrum games classics ★ ZX-81 releases — few and far between ★ Top Ten Turkeys</i>	
CHAPTER 6. Business	61
<i>Computerisation — when and why ★ Setting up ★ Software for the serious user ★ Case histories</i>	
CHAPTER 7. Machine Code	67
<i>Bits and bytes ★ Addresses ★ Registers ★ The stack pointer ★ The program counter ★ Instruction set ★ Glossary of machine code instructions</i>	
CHAPTER 8. Language	75
<i>Forth ★ Micro-Prolog ★ Logo</i>	
CHAPTER 9. Books	87
<i>Machine code ★ Information technology ★ Languages ★ Whodunnits ★ General interest ★ QL ★ Humour</i>	
CHAPTER 10. Starter Pack	93
<i>Seven simple programs with line-by-line explanations</i>	
CHAPTER 11. Program Printout	103
<i>Twenty-five pages of new games and utilities for the ZX-81, 16K and 48K Spectrum</i>	
CHAPTER 12. Reference	129
<i>Software Directory ★ Software publishers ★ Hardware Directory ★ Club Corner</i>	

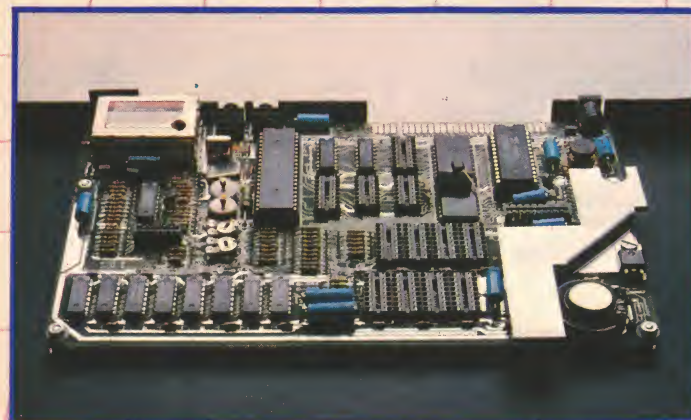
■ The **LO>>PROFILE** is a full size QWERTY keyboard with full-travel positive-action keys for super-fast entry of programs and data. Presented in an ultra-modern and robust enclosure, the **LO>>PROFILE** is elegantly styled and is inclined forwards for convenient operation.

■ Advanced Memory Systems have established an excellent reputation for product reliability and prompt delivery. This product has been tested to an impressive 20,000,000 depressions per key and AMS offer a full one year guarantee.



Fitting your Spectrum computer board into the **LO>>PROFILE** keyboard is easy. All you need is a screwdriver to secure the board onto the base of the keyboard enclosure using the mountings provided. The **LO>>PROFILE** is compatible with all Spectrum add-ons.

If not available from one of our 200+ dealers then fill in the coupon and we will send you a **LO>>PROFILE** with our full no-quibble money-back guarantee. Or phone 0925 602690/62907 for instant Access or Visa orders.



Please send No. _____ **Lo>>Profile Keyboard** at £49.95 plus
£2.50 p&p (VAT inclusive)
I enclose a cheque/PO for £..... or debit my credit card.

Card No

☐ Barclaycard
☐ Access card

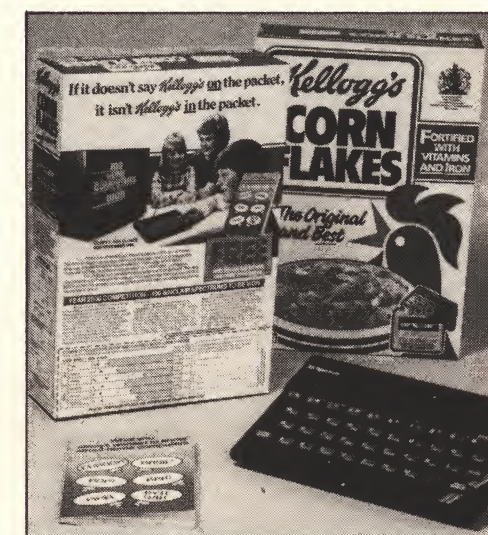
Signature _____ Date _____

Name _____

Address _____
_____ Postcode _____

To: Advanced Memory Systems Ltd.,
Woodside Technology Centre,
Green Lane, Appleton, Warrington WA4 5NG

News



Future historians may well regard 1984 as a turning point in the home computer market. Infiltration of computers into British homes reached new heights and the increasing sophistication of software meant that users became less tolerant of second-rate programs.

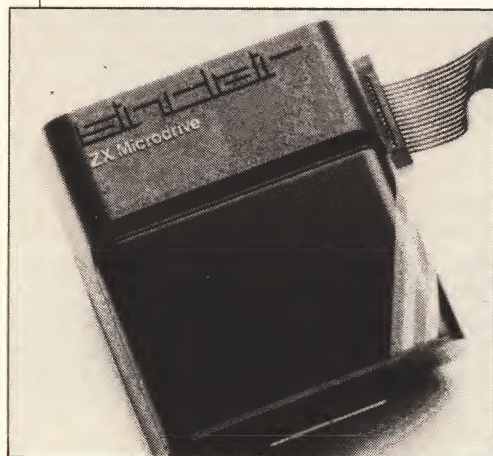
Many companies fell by the wayside and Sir Clive Sinclair unveiled his most ambitious hardware to date. Mike Johnston witnessed it all.

NINETEEN EIGHTY FOUR has been a year of both feast and famine, of glut and dearth, not only in the Sinclair world, but in the home micro world generally. It has been a year of silicon shortages and software surpluses. A year in which some microcomputers were late in appearing, others failed to appear and still others vanished completely.

It has also been a period of consolidation in the micro business when some of the smaller companies and some of the more colourful pioneers have had to give way to more established organisations and those sober-suited men with little knowledge of machine-code programming but a firmer grasp of the analysis of cash flow.

Chip shortage

First, the shortages. Apart from the most obvious shortages of QLs and in some areas Spectrums, not to mention the perennial microdrive cartridge problem, there was a general shortage of the basic building blocks of computer



and peripheral hardware during the year — the fabled microchip itself. Much of that was due to the improvements in the world economy — you didn't notice? — and everywhere, particularly in the US where they make a large proportion of them, the demand for chips rocketed as production increased.

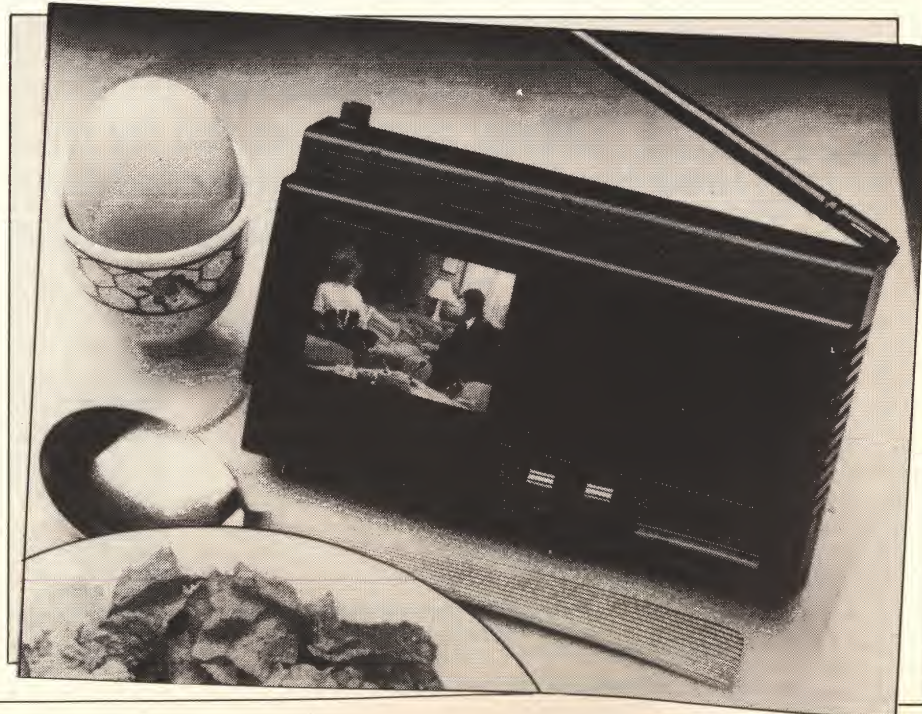
What that meant in practical terms was a slowing down of the development of new products and a reduced availability of existing products for at least part of the period. It has also meant that the prices of hardware items have not dropped as rapidly as might have been expected. In some cases, notably the Interface 1 and the Sinclair pocket TV, they have actually gone up — though not necessarily for that reason. The shortage did provide a convenient excuse for delays of one sort or another — “We simply can't get the chips.” Words to that effect were reportedly heard around Cambridge at the time of the early QL delays.

The good news is that everyone has been building chip factories. And since those take around eighteen months to get into production there should be lots of the stuff around by next year. Our local hero has not been left out of this development. Sir Clive Sinclair announced during the year that he would be investing large sums in the new wafer-chip technology over the next few years which could mean cheaper and faster super-computers from the company before too long. Do not be over optimistic, however; one Californian

company lost millions recently trying to do the same thing.

Despite the shortages there has been a fair amount of new hardware released during the year from a number of manufacturers and new joysticks, interfaces and keyboards for the Spectrum have been appearing in ever increasing quantities.

One interesting development has been the appearance of fast mass-storage devices for the Spectrum. Cassette recorders are cheap and, thankfully these days, reliable but can be unbearably slow for users wanting to make serious use of their machine. The Sinclair microdrive system, used with the Interface 1, appeared in the shops in the early part of the year after a slow start and



limited mail order availability. Since then a number of other systems have appeared ranging from speeded-up cassette drives to full-blown disc drives. Some of them are expensive, compared with the cost of the Spectrum, but as they become more popular prices seem likely to fall.

Plans to expand production of microdrive cartridges were announced during the year but the failure of software companies in general to use the new medium has probably delayed expansion. That in turn has been related to the continued high trade price of cartridges which, the companies say, make them uneconomic except for more expensive programs — some business software has been issued on cartridge.

Not only the giants suffered. The Jupiter Ace, produced independently by the designers of the Spectrum, crashed in late 1983, and the Aquarius, a low-priced colour machine which once seemed a possible Spectrum competitor had to be rescued around the same time. The Dragon, which had earlier benefited from Spectrum shortages, was rescued a number of times in the course of the year before finally calling in the receiver in June. Oric announced that they would reduce their UK marketing and concentrate on overseas sales.

Record sales

All that was in spite of the record sales of home computers at the start of the year. Sinclair had sold over a million

chines and the allegation — denied by the company — that quality control was being sacrificed to meet the demand.

Micro sales generally were in the doldrums during the summer months but in the autumn Sinclair announced that Spectrum production would be raised to 200,000 units a month including overseas orders to be accompanied by a massive advertising campaign and promotional offers on Spectrums and Microdrives.

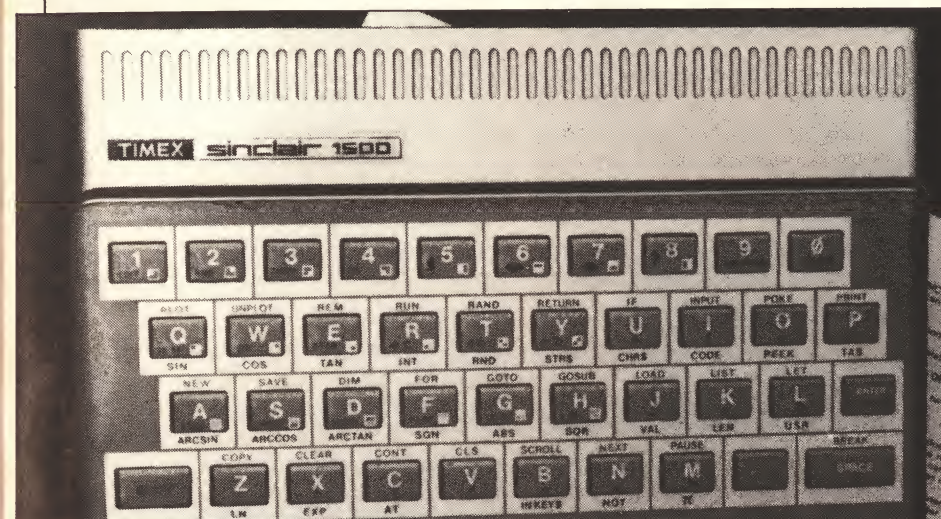
ZX Printer RIP

Not all Sinclair products have been wholly successful. The ZX Printer was phased out during the year and sales of Interface 2, which used ROM-based software and allowed the use of joysticks, appear to have been less than meteoric. The ROM software was a little expensive when first launched in late 1983 at around £15.00 and some items were later reduced to under £10.00. It seemed that customers were prepared to live without the instant program appeal of ROM software and the predicted take off of programs in that form never really got off the ground.

Software surplus

From hardware shortages to the software surplus. The year has not been bad for Spectrum software, although ZX-81 owners found new products a little thin on the ground. There have been many excellent new programs including Spectrum-stretching graphics games and adventures in *The Hobbit* tradition such as *Valhalla*, *Sherlock* and *Lords of Midnight* which have broken new ground, together with new languages and utilities like Micro-Prolog and even a C compiler.

The software business has moved unashamedly in the direction of the toy/leisure mass market with many pro-

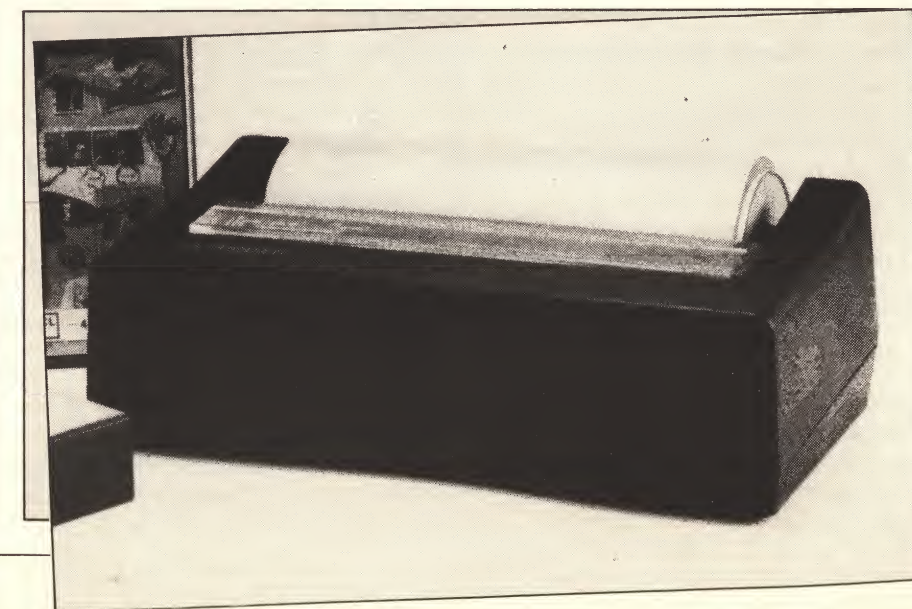


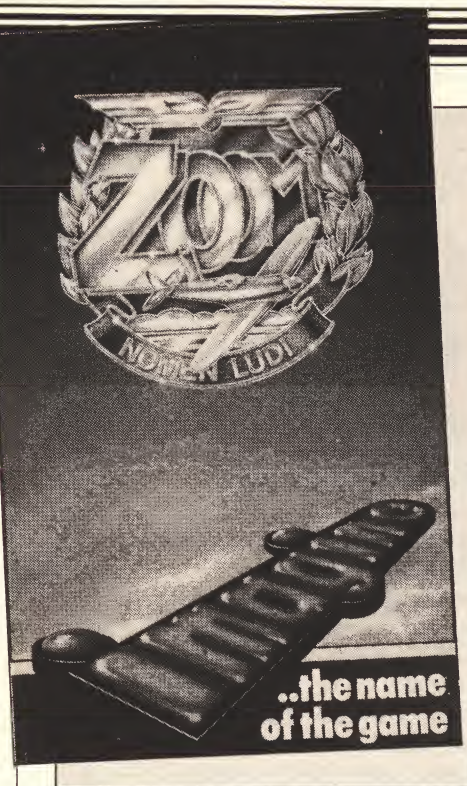
Cassettes have remained the favourite software medium and seem likely to remain so for the immediate future.

For a number of Sinclair's competitors the main shortage during the year has been one of sales. The year opened to the bellowing of wounded hardware giants and the squeaks of some smaller fry. Texas Instruments and Atari both made substantial financial losses at the end of 1983. Texas, a major chip manufacturer, was sufficiently convinced to pull out of the home micro business altogether, while Atari, one of the first companies in the home micro world, has been taken over by another company and is being restructured.

The continuing success of Sinclair in this country was undoubtedly a factor but the fierce price-cutting wars in the US played a major part. Sinclair was indirectly affected by that too when their US licensee Timex announced in February that they would be leaving the micro market and would no longer offer their versions of the ZX-81 and Spectrum in America.

Spectrums, and the same for the ZX-81, by that time and the public continued to buy them at the rate of 50,000 a month, rubber keyboard and all. Alas, that was still not enough to meet demand and shortages were reported around the country well into the year. More disturbingly there were also reports of relatively high failure rates of new ma-





grams linked to TV series, films, books and even comics. Many software companies have been negotiating to secure rights to cartoon characters, actors and sports personalities. Games software charts have become a fact of life and Mickey Mouse(TM) on the Spectrum cannot be far away.

More established companies have joined the software producers, including publishing houses like MacMillan, Penguin and the Daily Mirror group — Mirrorsoft — in all three cases with educational software, which is perhaps an extension of their existing activity. These developments and others, such as the sale of one of the best known software houses, Quicksilver, to a publishing group, will give some stability and financial support to what has been a mushroom growth industry. From the customer's point of view it should again mean a better standard of quality and reliability.

The Great Shake-out

Whilst the year may have been a good one for users it has been something of a nightmare for software producers. Many software houses had expected 1984 to be a bonanza year for software sales and shops had geared themselves up to meet the demand. However, the expected surge never came and apart from the glossy chart-topping programs which everyone bought, the ordinary bread-and-butter programs sold quite poorly.

Spectrum owners, especially new owners, found that a lot of very good software was being sold at under the

usual price by shops wanting to clear their shelves, but it was bad news for software houses who found themselves with bills to pay and little income to do it with. Some were able to tighten their belts and soldier on over the summer months, others quietly ceased trading or were taken over, and a few crashed spectacularly — notably Imagine Software who managed to hit the headlines right up to the end.

Various reasons were put forward for the software slump: it was said that too many similar programs were being produced by too many software houses — in other words, there was a lot of junk about — and that software was too expensive anyway at around £6.00 a program.

Pocket money games

At least two companies introduced a range of budget priced software at around £2.00 per tape in the spring and throughout the year others followed suit since the move was by all accounts fairly successful. It is unlikely that budget software will ever match the standard of the better quality programs since good software takes a lot of time and money to develop and a lot more to advertise and distribute.

Future developments may see a divergence between the cheap and cheerful on the one hand and the top quality products which may increase in price. One outcome is that computer-owners — and nearly half of these are Sinclair owners — have become more discriminating about the software they are prepared to purchase.

The other contributing factor to the loss of sales as seen by the software houses was the continuing problem of software piracy. The Guild of Software

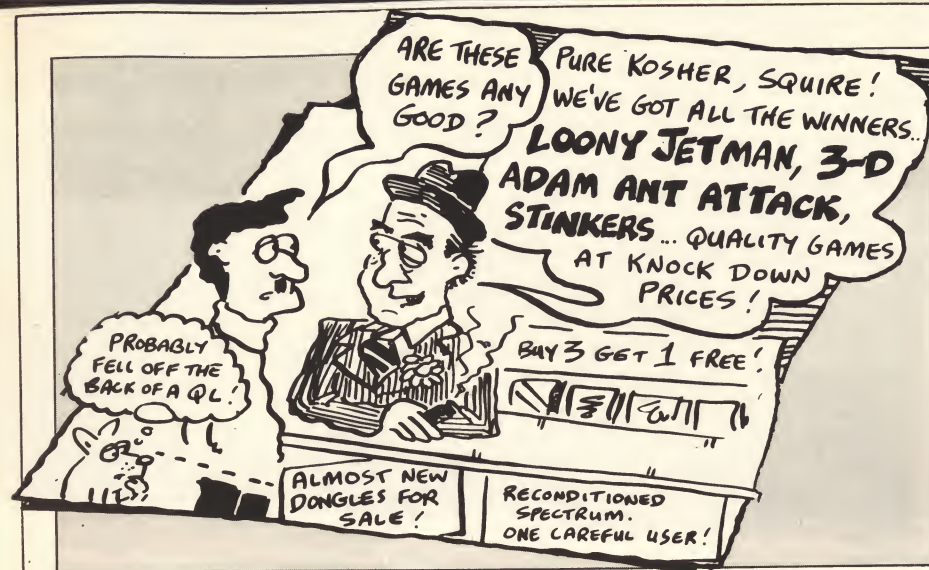
Houses (GOSH) set up a £50,000 fund during the year to help publishers to take legal action against commercial pirates. Another organisation, the Federation Against Software Theft (FAST), was set up to fight for changes in the law to help combat the same offenders. Home copying remained a problem and software companies have developed a variety of ingenious methods and routines to avoid this. But the totally unbreakable program is very difficult to produce. The director of one software company commented that it was little wonder that there had been a drop in program sales since there was so much 'free' software in circulation already.

The QL launch

Perhaps the most noticeable micro famine of 1984 and certainly the most newsworthy was the great QL shortage. That was unfortunate because in the end it obscured much of the real merits and achievements of Sinclair's new machine.

Rumours that Sinclair Research was about to launch a new computer circulated for most of 1983. One magazine went so far as to publish an artist's impression of what it might look like — which needless to say was nothing like the machine that did appear. There were stronger whispers around Christmas last year that a new 'professional' machine was in the pipeline with, it was hinted, the latest 16 bit technology but this "would not be available until much later in the year."

It came as a shock to quite a few pundits — though not to *Sinclair User* who was among the first with the news — when the QL launch came in early January with specifications which seemed to exceed the wildest dreams of



most Sinclair buffs — not a 16 bit but a 32 bit processor, 128K RAM as standard with a 500K add-on board on the way, a new structured Basic with procedures, multi-tasking, windowing, microdrives, huge hi-res screen... and if that was not enough a stunning suite of professional programs which would normally cost more than the whole machine. It was all too much.

At around £400 it was a touch pricey... but with those specifications who was complaining? Young Spectrum owners all over the country began to calculate how many years of extra dishwashing, lawn mowing and window cleaning it would take to raise the money. Bemused booksellers sold out of normally slow moving books on the Motorola 68000 without knowing why. Sinclair had done it again. Heavy duty computing was within reach of the common man.

Production problems

The let-down came more slowly. First deliveries were promised for the end of February, but seasoned Sinclair watchers treated that with a certain amount of caution. It soon became apparent that this date might slip a little. The reasons for the delay were not altogether clear — 'shortage of chips' it was said, 'very high demand,' a few teething problems with the Basic. At the end of February Sinclair wrote to purchasers apologising for the delay and promising first deliveries for the end of April.

During the following months the early ecstatic reception became a little soured. QL delays were reported in the national press, there were cartoons about it, jokes on TV. Was it really a 32 bit machine or just a dressed up 8 bit machine? And that keyboard — after all those complaints about the Spectrum

production problems are a frequent if unfortunate feature of many high technology products. A number of companies including some of the largest had experienced similar difficulties but without so much public comment. The problem for Sinclair was perhaps that wizards, electronic or otherwise, are expected to perform miracles and a lot of people were very very disappointed that he could not. The fact is that fewer people care about all the other products which are delayed whereas they care very deeply about Sinclair's. The company was slow to explain exactly why there was a delay with the QL and that did not help. People might have shown more patience if they had been taken



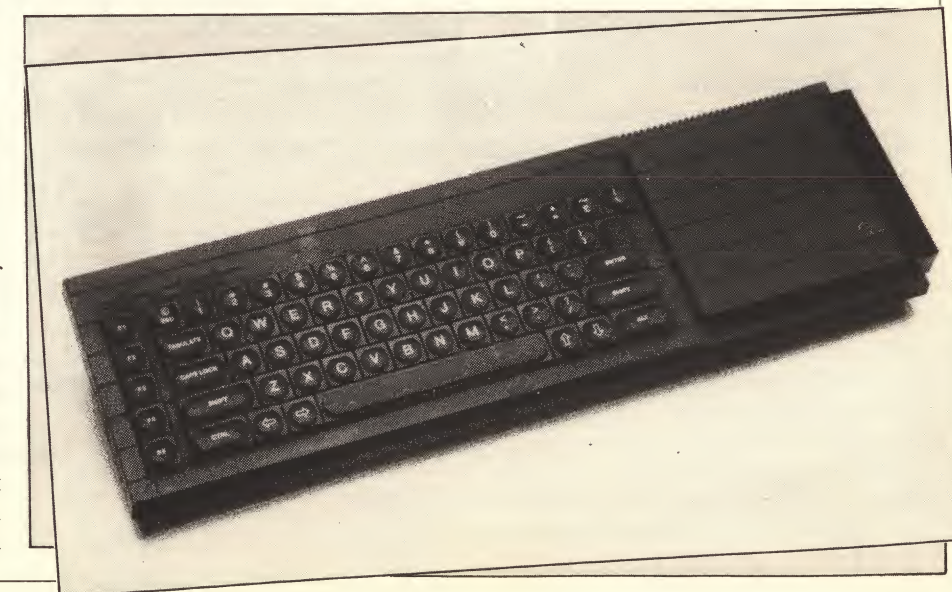
was it really another membrane keyboard?

Unfair criticism

Much of that was nit-picking but it did not help the machine's chances. At a later stage Sinclair protested, not without justification, about the level of criticism which he felt singled out his own company unfairly. He made no excuse for the delay but pointed out that late

into the company's confidence earlier.

In fairness to Sinclair it should be said that he did meet the revised schedule on time and first deliveries were made at the end of April as promised — if not quite in the condition promised. It appeared that one reason for the delay was that the operating system would not fit the original QL and an extra EPROM was supplied attached to the rear of the machine when delivered.



Further bugs appeared in the 'SuperBasic' which went through several versions as the machines were sent out. The early manuals had no section on Basic and other sections seemed to bear only an approximate relationship to the machine. There were problems with the packaged software... It was clear that a fair amount of 'tidying up' was needed.

What was not clear was why the company had launched so early a machine so far from completion. Among the theories advanced were that Sinclair was seeking the BBC contract, which came up for renewal in 1984, or that he was trying to steal a march on other competitors.

More unkindly there were allegations that the company was seeking to bolster its cash position to make it more attractive to investors for the planned share flotation later in the year. The company was quick to deny that, pointing out that it was a responsible company with

from major suppliers including Winchester disks, floppy discs, CP/M, BCPL and other languages calculated to warm the heart of the enthusiast — professional or not. At least one QL magazine was under way and many others carried QL pages. In addition to

the official QLUB support group an independent users group had been formed.

After a stormy beginning the machine at last seemed under way. For those with short memories it is worth recalling that the BBC machine which has enjoyed a certain amount of success, fell rather short of its initial specifications and was subject to possibly worse delays before finding its niche.

The future

One major change of strategy from Sinclair in 1984 was the move to full retail distribution of products and away from the traditional mail order sales techniques. It is not yet clear if that will apply to all new products, but it could mean the end of the 28 day waiting period as we have known it. Coupled with a large advertising campaign, including TV advertising for the first time, it suggested that the company was

seriously getting to grips with mass marketing and an improved service to customers. It almost certainly means there will be many more Sinclair users to swell the ranks and attract even more software and peripheral manufacturers to produce for them.

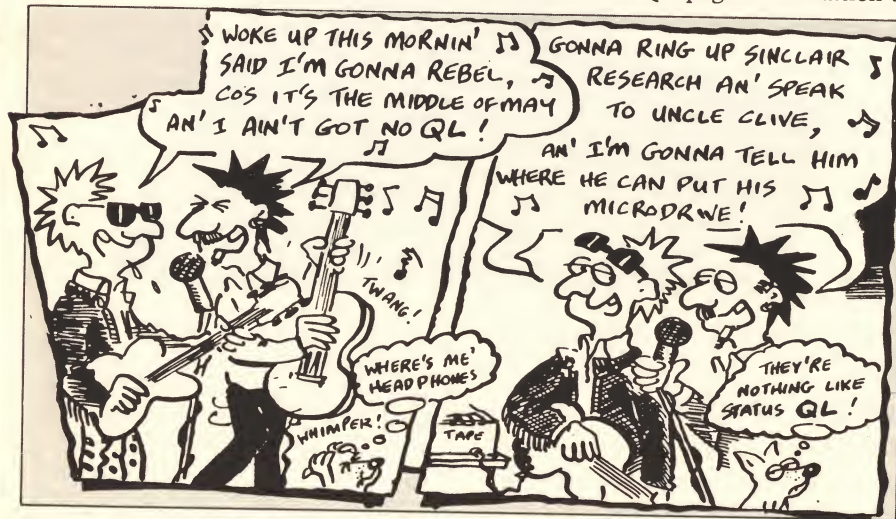
Sir Clive himself continued to expand his public image of popular electronic guru and entrepreneur at large with frequent appearances on radio and television. In the popular mind he attained a status somewhere between H G Wells and Sir Freddy Laker (as used to be). As well as home computer products he announced during 1984 the development of a low price satellite TV aerial and, more substantially, plans for his 'brains trust' Metalab to work on a fifth generation computer project to rival the Japanese efforts in the same field. Initial manufacture of his electric vehicle began during the year and sales are expected to begin in the New Year.

Sinclair users had little cause for worry over the future of their machines in 1984. While other manufacturers performed various kinds of nose dive during the year Sinclair maintained an overall lead in the home micro field and continues to dominate the market despite the odd slip. It is clear that he

intends to retain that lead.

The big question for 1985 is whether Sinclair — and other British manufacturers — can gird his loins and defeat the Japanese Armada of MSX machines now steaming towards our coast. He has had experience of Far Eastern competi-

tion before and there is little doubt that he takes it seriously. Whatever the event, every Sinclair user will wish him success in the knowledge that a low priced state-of-the-art ZX-85 is a more likely prospect from Cambridge than from Tokyo — even with the odd delay.

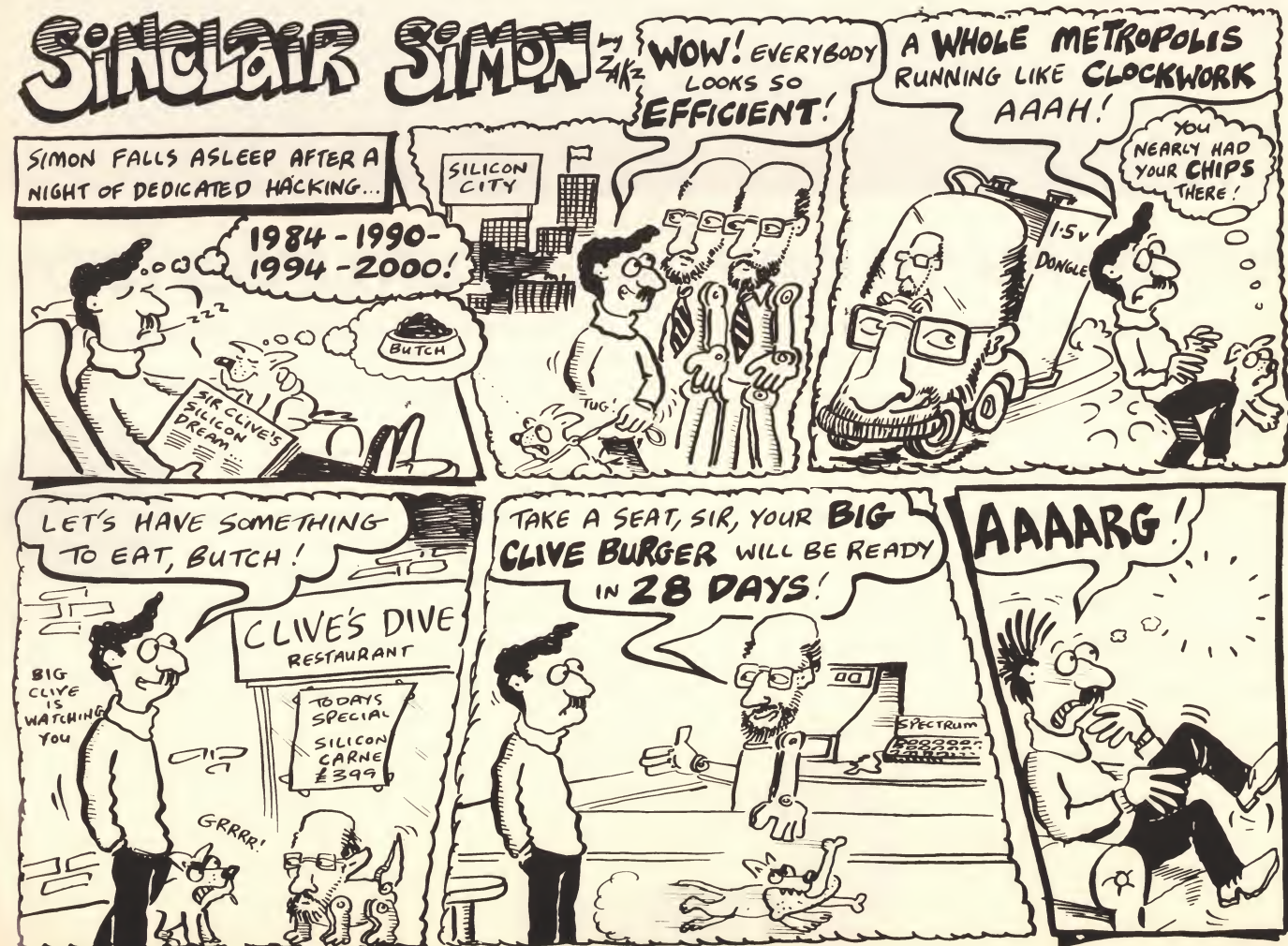
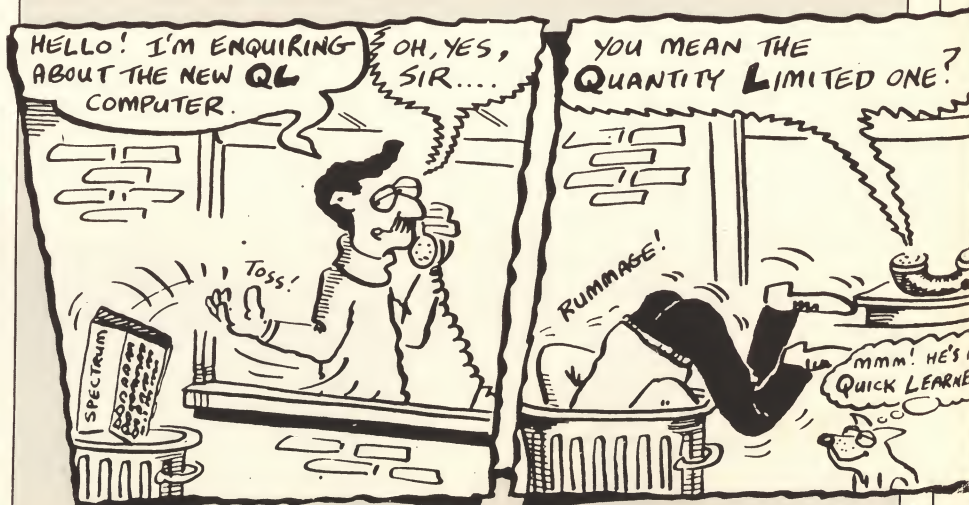


adequate finance and had no need for such stratagems. In the event, the public flotation has been delayed. Whatever the reasons for the problems the whole business was something of a dent for both Sinclair and for the QL.

In the shops

Despite all the fuss, most of the backlog in QL orders had been cleared by late summer and most of the major bugs were being ironed out of the software. QLs were on sale over the counter at the ZX Microfair in early September and full retail sales of the QL were due to start in the autumn. The Sinclair support industry was beginning to gear up with the production of books and the first hardware add-ons — extra memory boards, disc connectors and the rest.

Sinclair announced further ambitious plans for the machine with a host of major add-ons and software packages



TURBO CHARGE YOUR SPECTRUM



Outperforms any Spectrum interface

The unique Turbo interface from Ram gives you all these features – and more – in one unit:

- * A variety of interfaces including: Rom cartridges, two 9-way D plugs for standard joysticks, PLUS full expansion bus at rear.
- * Compatible with Kempston and Protek protocols.
- * Works with latest Quickshot Mk II auto rapid-fire joysticks!
- * Choice of Rom cartridge or tape cassette software.
- * Instant program loading with cartridge software.
- * Built-in power safety device – unique to Ram Turbo.
- * Full one year guarantee.
- * Immediate availability – 24 Hr despatch on receipt of P.O./credit card details (cheques – seven days).
- * Incredible value – only £22.95.

So don't wait around – simply complete the coupon and send it to us today.



Or call our credit card hot line on 02514 25252. (Access and Visa welcome).
Ram Electronics (Fleet) Ltd, 106 Fleet Road, Fleet, Hampshire GU13 8PA.

Please send me:

- _____ Spectrum Turbo Interface(s) at £22.95
+ £1 p+p (overseas orders £3 p+p)
- _____ Quickshot II Joystick(s) at £9.95
(Only when purchased with Turbo – normally £12.95 + £1 p+p)

I enclose cheque/postal order or charge my Access/Visa for: £ _____



Name _____

Address _____

Tel _____

SUA

To: Ram Electronics (Fleet) Ltd, 106 Fleet Road, Fleet, Hampshire GU13 8PA.

Trade and export enquiries welcome.

Chapter 2

Hardware World



With ever-increasing amounts of peripherals available for Sinclair computers, users are finding it more and more difficult to know what to buy.

We present a comprehensive buyer's guide to joysticks, keyboards and printers and review the best of the rest from the 1984 add-on market.

Joysticks

Joysticks are a popular addition to the Spectrum and many people find it essential for zapping aliens or flying a simulator. In the last year there have been a number of interfaces and joysticks launched, some with more success than others.

Nearly all joysticks have accepted a common standard, that of Atari which was the first company to add joysticks to a computer. The style of the joysticks may vary but the connection to the interface remains the same.

To use a joystick some form of interface is needed. One, or more, of three standards are normally used. 'Cursor', sometimes known as 'Protek', simulates the four cursor keys for the directions with 0 as fire, 'Sinclair' allows for up to two joysticks and simulates keys 1 to 5 and 6 to 0 respectively and 'Kempston' which does not simulate any of the keys but uses a port address. The most common of those is the Kempston but it is by no means universal.

To overcome that a programmable interface can be used which allows you to use any key for any direction or fire. The method of programming varies, some need a tape to be LOADED, while others require you to press a key and then to move the joystick in the direction which that key will simulate; some have physical connections which must be made to signify the key and others simply use a stored memory of which keys are to be used.

Grant Designs & EEC

Two designs combine the interface and joystick in one. The Grant Designs Spectrum Stick and a similar one from EEC clip to the Spectrum keyboard over the cursor keys and use a mechanical system to press the keys.

Inside the Spectrum Stick there are two frames which move independently up and down, and left and right; to those are attached ramps which push little feet onto the relevant keys.

The EEC joystick is more substantial. The casing is a one-piece moulding which fits snugly onto the Spectrum case. Inside that the stick is securely joined to a shaped plate which presses on one or more levers which in turn depress the key.

Both are priced at £9.95 and are far cheaper than anything else on the market. Provided they are used with restraint they are better than nothing.

RAM Turbo interface

The RAM Turbo interface from Fleet Electronics is what the Sinclair Interface 2 should have been and was not. As well as having two joystick ports, it has a ROM cartridge slot. Moreover, one of the joystick ports can be used for Kempston compatible games and the other can be used as a Protek compatible port.

It is similar in size to the Interface 2 but the rear connector is full Spectrum-size as opposed to the Interface 2 ZX-81 size. The area surrounding the joystick port is large enough to take a standard joystick connector. Users who have either built their own joystick or use the Flight Link joystick will have found problems with Interface 2 as the left-hand, most commonly used, port is too near the raised ROM slot.

In order to prevent users fitting the interface with the power on (the surest way to blow up your Spectrum) it incorporates a 'Spectrum Protect Adapter', a loop of plastic which covers the power socket on the Spectrum so that the lead must be removed before fitting.

If you are considering the Interface 2 then think again; the RAM Turbo interface is better designed and better value for money, costing £22.95.

Protek interface

The new switchable interface from Protek can be configured for all three of the standards by using a switch on the back. The interface is dead-ended, so it must be the last add-on fitted. As with the Turbo interface, it is liable to crash if used with a twin fire button joystick. Its main drawback is that in the Protek position the top row of keys is disabled, while in the Sinclair position only the 6,7,8,9 and 0 keys are disabled, which could make some games difficult to play. The switch would have to be moved, the key pressed, and the switch returned to its original position.

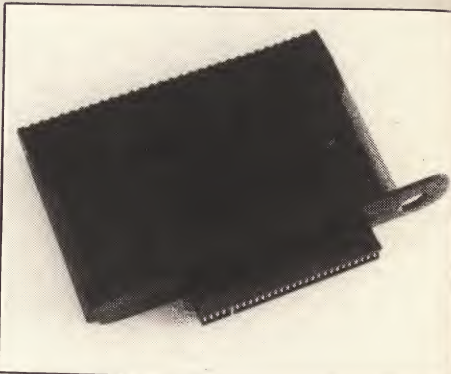
Despite the above problems, which should not affect most users, the interface is easy to use and covers most games on the market. It is priced at £19.95.

ELR interface

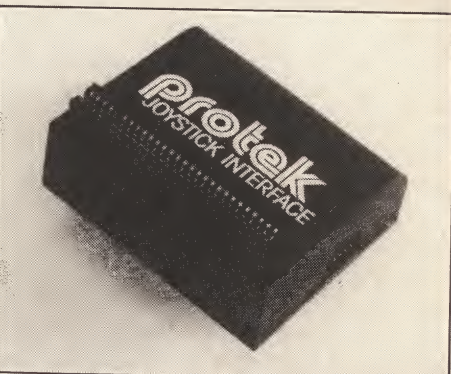
The programmable interface from East London Robotics for the Spectrum has the distinction of being the cheapest on the market at only £10.00, if bought



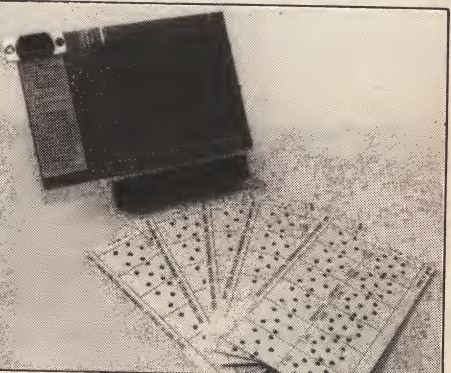
EEC



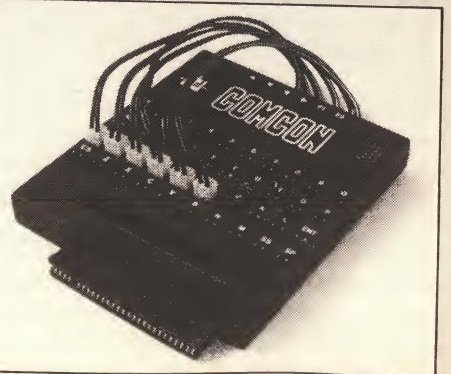
RAM Turbo



Protek



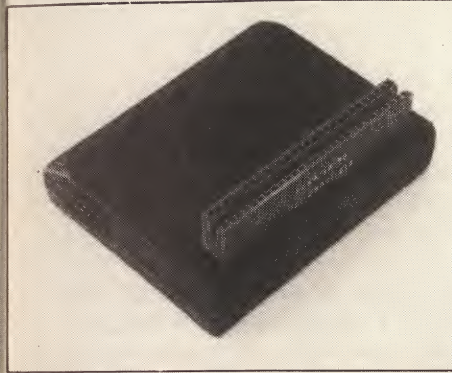
East London Robotics



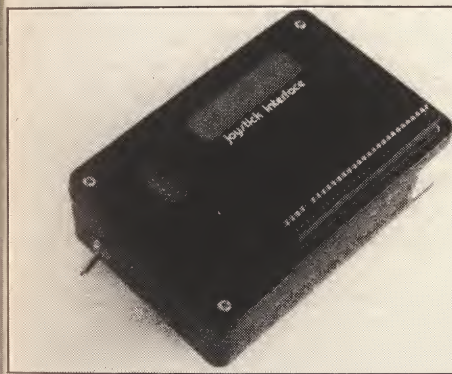
ComCon



Quickshot II



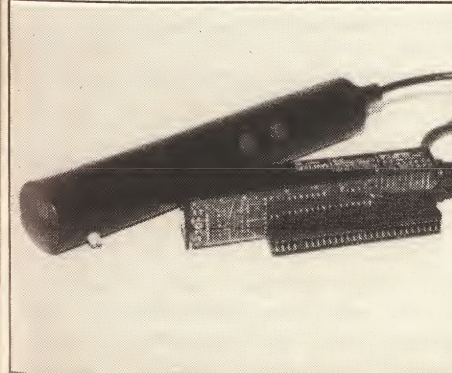
Custom Cables International



Page



Fox Electronics



Trickstick

Hardware World

together with a Trickstick.

The interface, as ELR would be the first to admit, was designed to be 'cheap and cheerful' and it is, even to the extent of being uncased. In order to program it small metal clips are used to connect pins which represent the various rows and columns of the keys on the keyboard. That is fiddly and time consuming if you need to reprogram it, but has the advantage that once programmed it is usable from power up. If the interface is purchased separately its price is £15.00.

ComCon interface

Frel Ltd market the ComCon programmable interface. Any key on the keyboard can be programmed and there is provision for two fire buttons. To match the interface Frel also markets two compatible joysticks.

The interface is of simple design and to program it you have to plug the relevant lead into the pin you want. That can be done with a program running and the keyboard is not disabled. Any joystick with a standard plug can be used and if that has only one fire button then the F1 lead is used. To allow for other add-ons there is an extender card which rises vertically from the front of the board.

The two joysticks which Frel markets are the Flightlink and the Quickshot II which has been adapted to have the independent fire buttons, F1 on the trigger and F2 on top.

At £19.95 the interface is one of the least expensive and the ease of programming should ensure steady sales. The two joysticks are £10.50 for the Flightlink and £13.95 for the Quickshot, both less £2.00 if ordered with the interface.

CCI interface

The interface from Custom Cable International needs a software tape to program it. It can accept any standard joystick; there is no extender card, so it must be the last add-on plugged in.

Programming the interface is straightforward. In all, only 10 key-presses are needed plus two to confirm everything is order. The program is written entirely in BASIC which makes it very easy to customise.

This is a good interface and competitively priced at £15.00, but it could have been even better if the instructions had been more detailed.

Page & Rainbow

The Page and Rainbow interfaces are of the kind where you must push a key, move the joystick, release the joystick

again for the four positions plus fire, again for the diagonals, again for the diagonals plus fire and finally the fire button on its own. All of that can be done with the game running as the keyboard can still be used. Some people will find that time consuming and fiddly.

The one really useful feature of the Page interface is a rapid fire switch. When that is in the up position it simulates the fire button being held down, very handy for the 'zap everything in sight' type of game. The Rainbow interface includes a beep amplifier.

With the Rainbow interface at £24.00 and the Page one at £26.00, they deserve consideration provided you do not mind the time and hassle of programming them.

Fox Electronics

The programmable interface from Fox Electronics, on the other hand, is probably the easiest of all to program. You only have to flick a switch once you have told it which keys to use.

When the switch is up a menu is displayed on the screen. You have the option of creating a new key set, from any of the forty keys or selecting, with a single keystroke, one of the sixteen sets already created. Pressing the E key exits to Basic to load the game and programs the joystick. If necessary the key sets can be saved on tape. Leaving the switch down enables the Spectrum to ignore interface.

Inside the interface is a RAM chip and a small ni-cad battery, and on power down it retains the information in its memory for a minimum of six months.

When switched on the interface pages the Spectrum ROM and jumps to the program held in its RAM. The program then transfers itself into the Spectrum RAM, pages the ROM back in and puts the menu on the screen. On pressing the E key the program transfers itself back into its own RAM, sets up the joystick and clears the Spectrum memory. Any new key sets created are therefore saved in the process.

All that sounds clever but it also has two beneficial side-effects. When the switch is down, the interface causes a hardware reset; in other words, if you have a game running you can jump out of it without pulling the plug — a saving on plug wear.

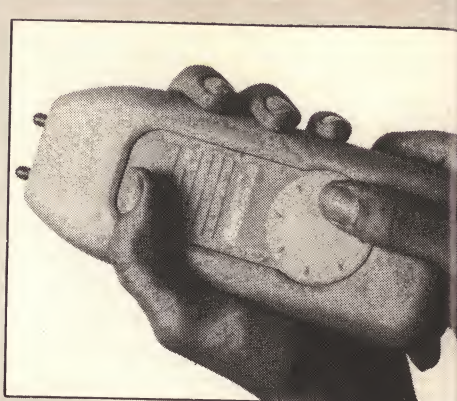
Secondly, details are available from Fox for a machine code programmer to adapt the interface as a pseudo ROM. Often used routines such as toolkits, printer driver programs or complete



Delta 3S



Super Champ



RAT

character sets could then be loaded at the flick of a switch. At £28.95 the interface is excellent value.

Trickstick

One of the more unusual joystick devices launched in 1984 was the long awaited Trickstick from East London Robotics. In fact, to describe it as a joystick is misleading. It is a 7in long black cylinder with six touch sensitive pads which simulate the four joystick movements, plus two fire buttons.

The Trickstick comes with an interface, up to eight of which can be plugged in at any one time, and has the option to give proportional movement on specially written games. It simulates the Kempston joystick so there are many games available on which it can be used.

The touch pads use your body as an aerial to pick up mains hum. That is converted by the electronics to simulate the key press. As some people are better aerials than others the sensitivity of the stick can be altered by a small screw near the top. A minor fault is that the lead from the interface to the stick fouls the power lead.

It takes some getting used to unlike a joystick, but once mastered is just as useful. As it has no moving parts it should stand up to wear and tear and it is quicker than most others. It costs

£34.50.

Nineteen eighty-four has been a quiet year for standard joysticks with only two given much publicity.

Delta 3S

The Delta 3S from Voltmace has two independent fire buttons, one of which is doubled up for left or right handed players, and rotary switches are used on it. Those switches give it a light action and it can be comfortably held in the hand. At £10.00 the joystick is good value, its solid construction should ensure long life.

Super Champ

The Super Champ, currently marketed by Dean Electronics, is unusual in that the cable is held in the base. For use the cable is pulled out and, when you have finished, rotating the stick winds the cable inside.

The stick is very long and thin and has two fire buttons, one on top and one in the trigger position; those both operate the same switch. The base is, as expected, larger than usual and has four rubber suckers underneath.

Because the stick rotates there has to be a certain amount of play in the pivot. That makes precise control difficult unless you hold down the centre. Despite that the joystick works tolerably and the convenience of not having the wire lying around almost makes up for it.

The Super Champ is priced at £12.95.

RAT

Finally, if the advertising is to be believed, the age of conventional joysticks is past and the future belongs to the RAT and its offspring. The RAT (Remote Action Transmitter) operates in a similar fashion to TV remote control handsets; you simply wave it in the general direction of the screen and press the touch-sensitive movement and fire buttons.

It is sold with a receiving unit that connects to the Spectrum and is compatible with most popular Spectrum software. The RAT is powered by a normal PP3 battery which fits into the underbelly of the handset.

The first of its kind, the RAT suffers minor teething problems and can cause excessive hand ache but, like the Trickstick, you will soon grow fond of it. The price is £29.95 from Cheetah Marketing and high street stores.

Grant Designs, Bank House, Reepham, Norwich, Norfolk NR10 4JJ. Tel: (0603) 870852.

EEC, 1 Whitehouse Close, Chalfont St Peter, Buckinghamshire SL9 0DA.

RAM Electronics (Fleet) Ltd, 106 Fleet Road, Fleet, Hampshire GU13 8PA.

Protek, 1A Young Square, Brucefield Industrial Park, Livingston, West Lothian, Scotland.

East London Robotics Ltd, Gate 1, Royal Albert Docks, London E11. Tel: (01) 474 4430.

Frel Ltd, Hockeys Mill, Temeside, Ludlow, Shropshire SY8 1PD.

Custom Cables International Ltd, Units 3 & 4, Shire Hill Industrial Estate, Saffron Walden, Essex.

Page Computing, 28 Burwood Grove, Hayling Island, Hampshire PO11 9DS.

Rainbow Electronics, Glebe House, South Leigh, Witney, Oxfordshire OX8 6XJ.

Fox Electronics, 141 Abbey Road, Basingstoke, Hampshire RG21 9ED. Tel: (0256) 20671.

Voltmace Ltd, Park Drive, Baldock, Hampshire S67 6ES.

Dean Electronics Ltd, Glendale Park, Fernbank Road, Ascot, Berkshire SL5 8JB.

Cheetah Marketing Ltd, 24 Ray Street, London EC1R 3DJ. Tel: (01) 833 4909.

Interface	Programmables	Cursor	Sinclair	Kempston	×2 Fire	Price £
Grant		✓				9.95
EEC		✓				9.95
RAM Turbo		✓	✓	✓		22.95
Protek		✓	✓	✓		19.95
ELR	✓				✓	15.00
ComCon	✓				✓	19.95
CCI	✓					15.00
Page	✓					26.00
Rainbow	✓					24.00
Fox	✓					28.95



SAGA 1 EMPEROR

THE KEYBOARD FOR ZX SPECTRUM COMPUTERS

- EASY FITTING
- PLEASURE TO USE
- LONG LASTING
- SELLING FAST

Saga 1 Emperor, equipped with 67 keys, has been carefully designed to incorporate the prime keyboard functions of the ZX Spectrum personal computer.

The style is easy: — for your benefit, the SAGA 1 Emperor Keyboard and housing retains the rear expansion dimensions and accessibility for compatibility with all Sinclair's own peripherals including interfaces and microdrive 1 and most other add-ons... including ours!

Not only this but we have ensured that the assembly of the keyboard is simple — and fast. No soldering is required, so that within just 5 minutes you can replace your current ZX Spectrum keyboard with the new SAGA 1 Emperor.

AND ALL THIS FOR JUST £54.95 (inc. VAT)

SOUNDBOOST

Hear that keyboard click... with a SAGA SOUNDBOOST

Your Spectrum's sound could be continuously adjustable from a whisper to a roar through your television. With our soundboost, no modifications need be made — just three easy push on connections. Supplied built and tested to fit in minutes with no previous experience required; we send full instructions for immediate use. Yours for Only **£9.49**



FLEXICABLE

Developed to relieve your Spectrum from the pressures of life. This neat 9 inch FLEXICABLE comes with two connectors which fit your Spectrum and your add-ons — taking the load with ease.

Only **£9.49**



Please write to: SAGA Systems Limited
Woodham Road, Woking, Surrey
Telephone Woking (04862) 69527/22922
or Telex 859298
all prices include VAT
P&P free for Sound Boost and Flexicable.
P&P £1.25 for SAGA 1 Emperor U.K.
£4.00 for SAGA 1 Emperor Europe

Please send me

Name

Address

The following SAGA PRODUCTS

Please make CHEQUES/P.O. Payable to SAGA Systems Ltd.



Keyboards

A COMMON complaint levelled at the Spectrum is the dead flesh feel of the keyboard. That has encouraged a large number of companies to produce full-sized add-on keyboards for it. We look at seven of the most popular keyboards on the market ranging from a simple replacement board to models of a professional standard.

In each case we fitted a Spectrum and Interface One to the keyboard to see how difficult that operation was and also attempted to fit the power supply according to the instructions or inside if there was room. The ease of fitting other add-ons was also considered and a Kempston printer interface was used to test that. Generally, any full-sized keyboard is a vast improvement over the original and of the seven all but the first can be recommended.

Two of the keyboards connect to the user port; the other five require you to open the Spectrum to fit them, and that will invalidate your warranty. It is possible that Sinclair will not notice that the case has been tampered with. We have yet to hear of anyone who has been refused a warranty claim.

Once the top of the keyboard has been removed the two ribbon cables connecting the old keyboard to the Spectrum are removed from their sockets and those from the new keyboard inserted. It is impossible to damage the Spectrum by fitting those the wrong way round, but care should be taken not to touch the circuit board unnecessarily. As with any piece of electrical equipment the power should not be connected while you are working on it.

The feel and size of a keyboard are important — after all, they are the reasons for buying a new one. The sizes are given in the table but the feel is more difficult to measure. The Saga and Lo>>Profile keyboards use keys which press down onto a membrane and have a small plastic dome to provide the spring, all the others use traditional switches. We suggest you try a keyboard before buying to make sure that it feels right for you.

K-Board

The first keyboard we looked at was the K-Board from Kelwood Computer Cases. That fits into the existing Spectrum case in place of the membrane keyboard. Fitting it requires major sur-

gery to the Spectrum. Once the two halves of the Spectrum have been separated the metal cover, which has the E-mode legends on it, has to be removed. The difficulty of that will depend on your Spectrum. After early complaints from Spectrum owners of the glue holding the cover in place melting due to the heat, Sinclair Research used a more powerful glue from Issue 3 onwards.

Next the old rubber mat and matrix are slipped out and the new PCB containing 40 moving key switches is put in its place. Onto that is placed a plastic housing, and over the top of everything goes the metal cover.

For each of the keys you are supplied with a new key cap, and each is hot foil printed in two colours, gold and red. Unfortunately, as the original keys were different in size there is a small gap at the side of all keys and the caps also overhang and shadow the legends on the metal cover. The method of printing the caps does not seem too durable as the legends could be removed by heavy scratching which tends to suggest that they would not wear well.

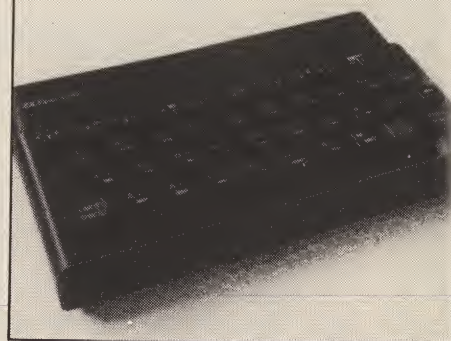
At £28.50 the K-Board is the cheapest replacement keyboard for the Spectrum on the market and as such is worth considering if you can not afford anything else. However, for your money you will get a keyboard that will probably invalidate your warranty as it would be difficult to restore the Spectrum to its original state, and which retains the same, cramped, layout and does not include a full-sized space-bar. Its only advantage is that as it retains the original case there is no problem fitting add-ons.

Mapsoft

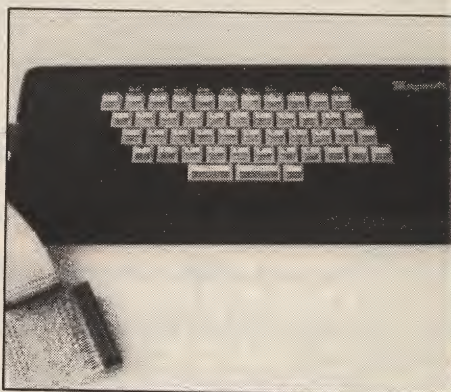
The Maplin keyboard, marketed under the name Mapsoft, is supplied in either kit form or assembled. It has an extender PCB that fits to the Spectrum and ribbon cable to the keyboard. The Spectrum remains outside the keyboard and so add-ons can be fitted with ease.

In all there are 47 keys, the normal 40 plus Delete, Graphs Lock, Shift Lock, Caps Lock, single key E Mode and two space keys. There is also one extra key that you can wire up to your own requirements.

Assembling the kit is straightforward provided you are experienced at soldering. None of the chips is socketed and special care has to be taken with them.



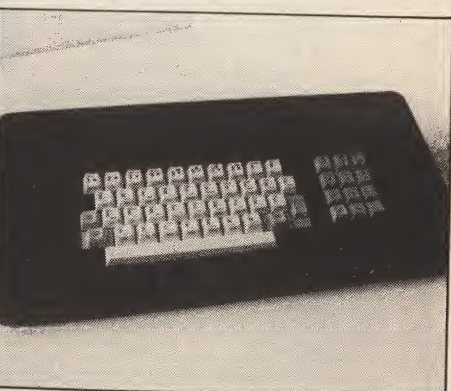
K-Board



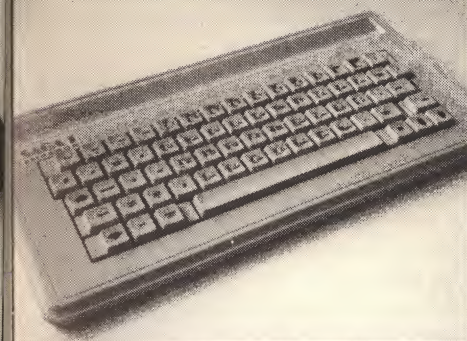
Mapsoft



dk'tronics



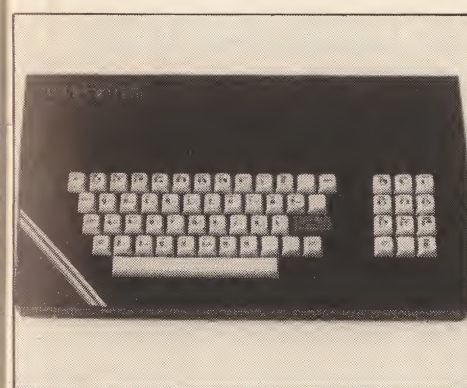
Lo>>Profile



Emperor



Stonechip



Transform

	Length in mm		Number of Keys	Size in mm	Price
	Q-P	Space			
Spectrum	195	N/A	40	230 x 140 x 30	—
K-Board	190	N/A	40	230 x 140 x 55	£28.50
Maplin	185	75	47	370 x 165 x 50	£44.95
dk'tronics	200	N/A	52	350 x 245 x 70	£45.00
Lo>>Profile	190	155	53	440 x 220 x 50	£49.95
Saga	190	150	67	370 x 180 x 50	£54.45
Stonechip	180	145	44	—	£59.95
Transform	188	150	60	410 x 220 x 75	£69.96

The length measured across the Q and P keys is the overall distance and is included to give an idea of the pitch. The original keyboard has been included for reference.

Once assembled the board is then fitted into the thin plastic moulded case, the top of which is held in place by pieces of Velcro. That top has the colour legends for keys 1 to 7 and 0 printed on it. The legends for the keys are three-colour cards that fit under clear plastic tops.

The keyboard offers a number of useful facilities at a reasonable price but it will always look like a DIY kit rather than a professional keyboard. The kit is priced at £44.95 inc.

dk'tronics

One of the best-known names for add-on keyboards is dk'tronics. Although its keyboard design has remained largely unchanged the company has made improvements to it. Those are the inclusion of a full-sized space bar, the addition of single key entry Delete and Full Stop and extra holes in the back of the case to allow the fitting of an Interface One.

The company estimates that 10 percent of all Spectrum users own one of its keyboards, a very sizable share of the market and to celebrate that dk'tronics has been offering four software tapes, worth £25.00, free to purchasers, during the last quarter of 1984.

The keyboard has the usual forty keys plus a separate numeric pad, which also incorporates the Delete and Full Stop keys. The three-colour legends on the keys have to be stuck on by the user but provided care is taken they will last a long time.

To fit the Spectrum into the keyboard it must be taken completely out of its case and screwed to the new base; if you have an Interface One then that is fitted first and the Spectrum PCB is slotted into it and then screwed down. As the PCB is fitted in the back right-hand corner there is an extender PCB for the microdrive lead which connects to Interface One by the solid connector supplied with the microdrive. The lead

is taken out of the case through a small slot in the left hand side. An extra long lead can be supplied if required.

On the left-hand side are three pillars on which you can mount the power supply. To do that the power supply must be taken out of its case and should therefore only be fitted by people who have some electrical knowledge. There are a number of problems concerning the fitting of the power supply; the Sinclair power supply has gone through a number of design changes, and if it is marked 'Made in Portugal' then it will probably not fit onto the pillars.

The design of the pillars is such that they will either need cutting to size and the power supply screwed down using oversized washers or it will have to be wedged in by use of a piece of foam, both methods slightly unsatisfactory. The mains side of the power supply is lead out of the case under, and very close to, the extender PCB. It would have been far simpler to put the Spectrum PCB on the left hand side in the first place and so avoid having leads crossing inside.

The Kempston interface will not fit unless you are using Interface One or are prepared to remove the cable's strain relief bar. It is possible to trim the case.

Despite those problems the keyboard is very popular. It has a nice feel to it and on the whole can be recommended. It costs £45.00 plus £1.25 p&p.

Lo>>Profile

The Lo>>Profile keyboard from Advanced Memory Systems Ltd is one of the newer keyboards on the market and looks very attractive. It is also very large, the largest of the keyboards reviewed, and that could be a problem if you are short of space.

As well as the standard Spectrum keys it has a separate numeric keypad, with extra Caps Shift and Full Stop keys, and there is an extra Caps Lock key included with the main keys, but those last two are not single key entry. It has a full-sized space bar. All the keys are sculptured and printed in three colours.

The Spectrum PCB has to be removed from its case and there are four bolts sticking up from the base of the keyboard on which it fits. In the review model they did not line up correctly and had to be moved by enlarging the mounting holes slightly. There are plastic washers on those bolts to protect the PCB.

While the base seems flimsy, when everything is bolted together it is held securely. Metal inserts in the top where

the base is bolted through add to the solidity.

When fitting Interface 1 the two long screws are removed and the interface then sits underneath the keyboard. That raises the keyboard to a better working angle. The bolts which secure the Spectrum PCB to the base are also passed through the interface. On the review model the size of the case meant that the normal microdrive lead was not long enough to clear it and so the drive had to be angled out backwards from the case.

As with most full-sized keyboards the Kempston interface was a problem to fit due to the increased height of the keyboard at the back. The best remedy for that would be to purchase a flexible connector which fits the user port.

Priced at £49.95 plus £2.50 p&p, the keyboard's good looks and positive feel will attract many users; its lack of any single key functions may, however, prove a drawback.

Emperor

The Emperor keyboard from Saga Systems is similar to the Lo>> Profile keyboard in that the keys use the same method of operation and it looks very good. What sets it apart is its remarkable 67 separate keys.

As well as the usual 40 keys there are 21 of the most commonly used functions on separate keys plus an extra three Symbol Shift and one Caps Shift keys. Those are arranged so that the functions are adjacent to a shift key. In that way just one finger can be used to hit both keys, a system that works remarkably well. Those extra functions are the mathematical — plus, minus, multiply, divide and equals, punctuation — full stop, comma, semi-colon and colon; plus hash and dollar — all operated by a symbol shift. Delete, Edit, Graphics, Caps Lock and the four cursor keys are operated by a Caps Shift. Two of the Shift keys are together for Extended mode. For good measure there are also two adjacent Enter keys, extra Run, Save and 0 keys and a full size Space Bar with a levelling device.

To fit the keyboard you remove the top half of the Spectrum and the bottom half, containing the PCB, is bolted under the base of the keyboard. The board with the actual keys is screwed to that and the top then screws to the base.

The whole process only takes about five minutes, as claimed in the advertisements, though you then have to stick the three-colour labels onto the keys which takes considerably longer. Pro-

vided care is taken, fitting the labels is not a problem and they stand up well to some heavy use. As the original Spectrum base is now the base of the new keyboard, add-ons such as Interface 1 connect without difficulty but the usual problems with the Kempston interface will be encountered.

The price of £54.45, plus £1.15 p&p, is slightly above average but the multitude of keys ensures value for money.

Stonechip

The Stonechip Electronics keyboard is one of the few keyboards where the Spectrum does not have to be opened to be fitted. For good measure it also incorporates a beep amplifier, Load/Save switch and a reset facility.

Fitting the keyboard is simple; the top half, which houses the keys and amplifier, is connected by a ribbon cable to a long, thin PCB. That has the aerial, MIC and EAR plugs on one side and sockets on the other. That is plugged into the back of the Spectrum and the keyboard screwed together, with the Spectrum inside.

The keyboard has 44 keys printed in three colours, the usual 40 including a full-sized 145mm, space bar plus a single key entry Delete and E mode. In addition there are two Reset keys which when pressed at the same time will reset the computer—particularly useful when with many games the only way to do that is to pull the plug which eventually leads to it becoming loose. You have to be careful as resetting will clear out the memory and as one reset key is placed next to the Caps Shift and the other next to the 0 you must get into the habit of using the Delete key rather than the normal method of using those keys, otherwise you may delete more than you intend.

At the top of the keyboard is the Load/Save switch which in its middle position acts as the beep amplifier. There are both tone and volume controls and a good clear sound can be obtained. If you are not using the amplifier then it is best to put the switch to one of the other positions as otherwise it picks up hum from the computer. Also on top there is an LED to show you that the power is on.

For some reason best known to Stonechip the company has repositioned the MIC and power sockets. The MIC socket is 5mm nearer the user port and the power socket placed between the two. That small change means that a number of add-ons will not fit. Interface 1, however, will fit with a slight strug-

gle as will the Kempston interface.

Priced at £59.95 the keyboard is one of the more expensive ones available but the ease of fitting and the facilities it offers make it a good buy.

Transform

The last of the keyboards reviewed is that from Transform. It is aimed at professional users as it is the only keyboard reviewed with a metal case and also one of the few with three-colour printed key caps.

There are 60 keys, including a numeric pad that has an extra Caps Shift and single key entry Full Stop plus a full size space bar. The other extra keys are Symbol Shift, Edit, Delete, E Mode, Colon, Semi Colon, Comma and another Full Stop — all single key entry.

Fitting is straightforward; the base of the Spectrum is retained as an insulator and bolted through to the base of the keyboard. If Interface 1 is used the two long mounting screws have to be removed and the bolts then go through it and the Spectrum. There is provision to put the power supply inside the case and special connectors are used to wire an on/off switch and a LED. That is relatively easy and you need not worry about the leads shorting out as everything is well insulated, although Transform do recommend fitting an extra earth wire for safety.

The keyboard has a solid feel to it and is certainly the best of the keyboards reviewed. The only problem that may be experienced is that with some Issue 3 Spectrums the ULA chip, if not marked with the word Ferranti, is working to the limits of its specification which may make the computer rather temperamental when used with the keyboard. Transform sell a buffer, at £6.95 inc, which cures that problem.

The price of £69.95 reflects the sort of use to which this keyboard will be put but it is certainly the best keyboard of the moment.

Kelwood Computer Cases, Downs Row, Moorgate, Rotherham S60 2HD.
Maplin Electronic Supplies, PO Box 3, Rayleigh, Essex SS6 8LR.
dk'tronics, Unit 6, Shire Hill Industrial Estate, Saffron Walden, Essex CB11 2AQ. Tel: (0799) 26350.

Advanced Memory Systems Ltd, Green Lane, Appleton, Warrington WA4 5NG.
Saga Systems Ltd, Woodham Road, Woking, Surrey.

Stonechip Electronics, Brook Trading Estate, Deadbrook Lane, Aldershot, Hampshire GU12 4XB. Tel: (0252) 333361.

Transform, 41 Keats House, Porchester Mead, Beckenham, Kent. Tel: 01-658 6350.

FOX YOUR SPECTRUM

WITH THE
ULTIMATE IN JOYSTICK INTERFACES
"IS VERY CLEVER AND PROVIDES THE EASIEST INTERFACE ON THE MARKET TO USE"
SINCLAIR USER FEBRUARY 1984



- ★ Complete compatability with all programs.
- ★ In-built memory to store up to 16 different keysets at one time.
- ★ Battery back-up so no loss of memory after power off (battery recharged during use so no replacements are required).
- ★ Full casing.
- ★ Full through port for further expansion.
- ★ Proven compatability with the Microdrive.
- ★ Built in pseudo ROM facility with room for personal tool kit.
- ★ Compatible with all Atari type joysticks.
- ★ Compatible with Quickshot II Rapid Fire.
- ★ One switch only for simplicity of use.

ONLY
£28.95

NO NEED TO REPROGRAM FOR EACH GAME JUST FLICK THE RESET SWITCH AND SELECT A NEW GAME KEYSET FROM THE MENU. IT'S SO SIMPLE, NO TAPES TO LOAD, NO LEADS, IT'S ALL IN ITS MEMORY!!

SPECIAL OFFER
OR BUY THE INTERFACE ABOVE WITH THE
QUICKSHOT I JOYSTICK ONLY £32.95 OR QUICKSHOT II £36.95

SPECTRUM UPGRADES

(ISSUE 2 AND 3)

16K to 48K
THE KIT REVEIUED AS "THE EASIEST TO FIT"
SINCLAIR ANNUAL
Full instruction supplied with this simply inserted kit, a high quality kit at a LOW! LOW! price.
(Please state issue 2 or 3 when ordering).
ONLY
£23.00
INCL.

NEW! NEW! NEW! SPECTRUM KEYBOARD

THE CHEETAH 68FX1, AN ALL NEW VERY HIGH QUALITY PRODUCT WITH

- ★ Full size querty keyboard ★
- ★ Numeric pad ★
- ★ Compatible with Microdrive ★
- ★ Accepts Interface 1 and Interface 2 (without modification) ★
- ★ Simple to fit ★
- ★ Sixty eight keys ★

A SUPERB NEW KEYBOARD ONLY £59.95 INCLUSIVE

SEND NOW TO:



FOX ELECTRONICS LTD
FOX HOUSE
35 MARTHAM ROAD
HEMSBY
GREAT YARMOUTH
NORFOLK NR29 4NQ
TEL: 0493 732420

PLEASE SEND ME
INTERFACE ☐
QUICKSHOT I + INTERFACE ☐
QUICKSHOT II + INTERFACE ☐
UPGRADE ☐
KEYBOARD ☐

Name
Address

Barclaycard/Access Number
[] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] [] []



THE DECISION to buy a printer to complete your computer system will be one of the most difficult and certainly one of the most expensive which you as a Sinclair user will have to make. The variety of printers available tends to confuse rather than facilitate your choice and many of them are likely to cost more than the rest of your system.

Yet if you are intending to use a Sinclair machine for something other than games playing, you will, sooner or later, need to make a permanent copy on paper. With that in mind *Sinclair User* looks at 11 printers, from the inexpensive to those suitable for small businesses.

Printers

THE FIRST consideration must be the purpose for which you intend to use a printer. All the printers reviewed will produce listings of programs, some faster than others and some more legibly. If you intend to do some word processing the type-style and method of printing is also important.

Most of the machines reviewed are dot matrix printers. That means that, just as characters on the screen are composed of a number of dots — pixels — the printer has a number of little pins which are pushed on to the paper, through the ribbon, to form the character. The greater the number of dots the better the definition of the character.

Just as user-defined graphics are made by changing the arrangement of the dots, dot matrix printers can easily change the type-style or the character. The printer holds those definitions in ROM and by telling it which set you want to use it can print in different type-styles and use different national characters, in some cases even Japanese.

Some of the printers use different methods. The Alphacom 32 and the Brother EP44 are thermal printers. That means they use special, more expensive paper although they are still dot matrix types. The pins become hot and cause the paper to change colour. The Brother can also use plain paper with the addition of a special ribbon.

The Dyneer DW16 is a daisywheel printer, using a wheel which has each character, like those on a typewriter, at the end of a spoke. To change type-styles the wheel has to be changed.

The Olivetti JP101 has a small cartridge which fires an ink-jet at the paper. As there are few mechanical parts the printer is quieter than most.

The Tandy CGP-115 has four ball-

point pens which form characters by moving the pen horizontally and the paper up and down.

The speed of a printer normally is measured in characters per second; in theory the higher the number the greater the speed. To test that we used a standard body of text, the first chapter of Genesis, and timed how long it needed to print. The text was prepared using Tasword 2 and printed via a Kempston E interface, where suitable, a total of 796 words over 77 lines.

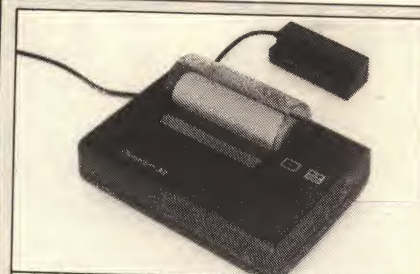
The results were surprising. The Microline 80, with a quoted speed of 80 cps, took nearly twice as long as the Mannesmann Tally MT80 and three times as long as the Olivetti. The answer would be to take such quoted figures only as a very rough guide.

Another thing to consider is how you want to drive the printer. Two of the machines reviewed connect directly to a Sinclair computer, the Alphacom 32 and the Seikosha. While that is undoubtedly the cheapest method, when and if you upgrade your computer you are left with a printer you may not be able to connect to your new machine.

There are two standards by which printers normally are driven, Centronics and RS232. Roughly speaking, Centronics interfaces have eight wires so that the bits of each byte travel to the printer at the same time along each wire. Serial interfaces have one wire and the bits travel along it one after the other.

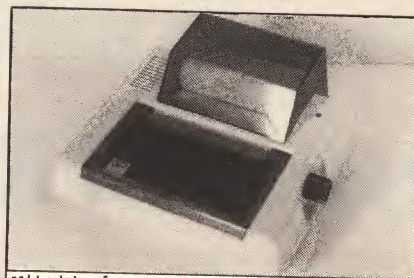
Whether you use Centronics or RS232 does not make much difference so long as you do not try to mix them. On many of the printers Centronics is fitted as standard and RS232 is available as an add-on at extra cost.

Your future plans must be taken into consideration. Users with a Microdrive



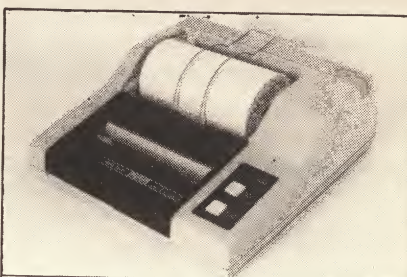
Let the earth bring forth grass, the tree yielding fruit after his kind in itself, upon the earth: and it was so. brought forth grass and herb yielding seed the tree yielding fruit, whose seed was in itself: and God saw that it was good. and the morning were the third day. Let there be lights in the firmament of the day from the night; and let them be seasons, and for days, and years: be for lights in the firmament of the earth: and it was so. two great lights; the greater light to rule the day and the lesser light to rule the night: he made the

Alphacom 32



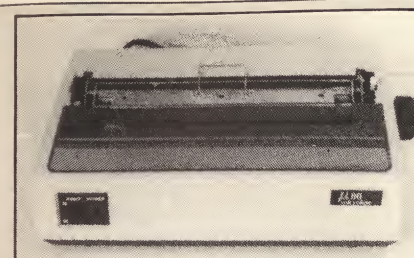
called he seas: and God saw it was good. Let the earth bring forth grass, the tree yielding fruit after his kind in itself, upon the earth: and it was so. brought forth grass and herb yielding seed the tree yielding fruit, whose seed was in itself: and God saw that it was good. and the morning were the third day. Let there be lights in the firmament of the day from the night; and let them be seasons, and for days, and years: be for lights in the firmament of the earth: and it was so. two great lights; the greater light to rule the day and the lesser light to rule the night: he made the

Seikosha GP50S



said, Let the earth bring forth grass, the herb seed, and the fruit yielding fruit after his kind: and it was so. the earth brought forth herb yielding seed kind, and the tree

Tandy CGP-115



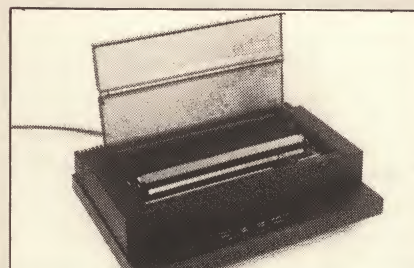
Let the earth bring forth grass, the tree yielding fruit after his kind in itself, upon the earth: and it was so. brought forth grass and herb yielding seed the tree yielding fruit, whose seed was in itself: and God saw that it was good. and the morning were the third day. Let there be lights in the firmament of the day from the night; and let them be seasons, and for days, and years: be for lights in the firmament of the earth: and it was so. two great lights; the greater light to rule the day and the lesser light to rule the night: he made the

Microline M80



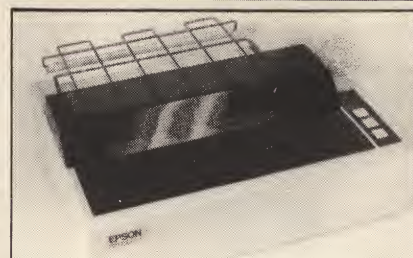
Let the earth bring forth grass, the tree yielding fruit after his kind in itself, upon the earth: and it was so. brought forth grass and herb yielding seed the tree yielding fruit, whose seed was in itself: and God saw that it was good. and the morning were the third day. Let there be lights in the firmament of the day from the night; and let them be seasons, and for days, and years: be for lights in the firmament of the earth: and it was so. two great lights; the greater light to rule the day and the lesser light to rule the night: he made the

Brother EP44



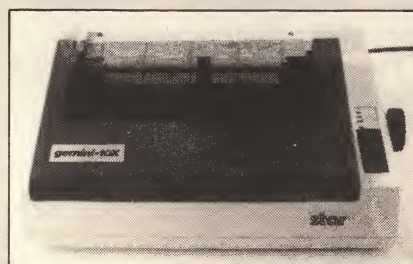
Let the earth bring forth grass, the tree yielding fruit after his kind in itself, upon the earth: and it was so. brought forth grass and herb yielding seed the tree yielding fruit, whose seed was in itself: and God saw that it was good. and the morning were the third day. Let there be lights in the firmament of the day from the night; and let them be seasons, and for days, and years: be for lights in the firmament of the earth: and it was so. two great lights; the greater light to rule the day and the lesser light to rule the night: he made the

Olivetti JP101



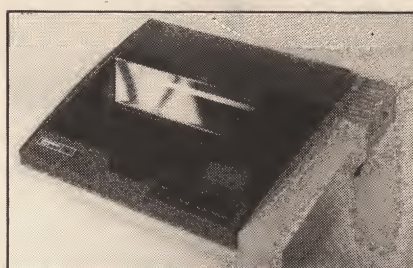
Let the earth bring forth grass, the tree yielding fruit after his kind in itself, upon the earth: and it was so. brought forth grass and herb yielding seed the tree yielding fruit, whose seed was in itself: and God saw that it was good. and the morning were the third day. Let there be lights in the firmament of the day from the night; and let them be seasons, and for days, and years: be for lights in the firmament of the earth: and it was so. two great lights; the greater light to rule the day and the lesser light to rule the night: he made the

Epson RX80



Let the earth bring forth grass, the tree yielding fruit after his kind in itself, upon the earth: and it was so. brought forth grass and herb yielding seed the tree yielding fruit, whose seed was in itself: and God saw that it was good. and the morning were the third day. Let there be lights in the firmament of the day from the night; and let them be seasons, and for days, and years: be for lights in the firmament of the earth: and it was so. two great lights; the greater light to rule the day and the lesser light to rule the night: he made the

Star Gemini 10X



Let the earth bring forth grass, the tree yielding fruit after his kind in itself, upon the earth: and it was so. brought forth grass and herb yielding seed the tree yielding fruit, whose seed was in itself: and God saw that it was good. and the morning were the third day. Let there be lights in the firmament of the day from the night; and let them be seasons, and for days, and years: be for lights in the firmament of the earth: and it was so. two great lights; the greater light to rule the day and the lesser light to rule the night: he made the

Mannesmann Tally MT80

can use the RS232 port fitted. If you are thinking of buying a QL it also has an RS232 port fitted as standard. Other add-ons such as modems also tend to use RS232 and so the system has advantages, despite the extra cost.

The width of paper the printer can handle may also be a consideration. Normal listings may use only 32 columns but if you like to add REM statements at the side or for word processing, 80 is the minimum.

It is often necessary to indicate to the printer what you want it to do, whether changing the type-style or feeding paper through it. There is a standard for it known as ESCape codes, i.e., codes which escape from normal printing to do something different. Those codes were originated by Epson and are now used by most other makes of printer.

To help you each printer is now considered, briefly, in turn. The Sinclair printer has not been included as it has been discontinued and will be available only while stocks last.

Alphacom 32 is the cheapest on the market at the moment. It has reasonable print quality and is easy to use. It has only a ZX-81 size connector, so Spectrum users must use it as the last add-on. From Dean Electronics or Prism.

Seikosha GP50S is an inexpensive plain-paper printer, not particularly fast and rather noisy but easy to use. It has a Spectrum-size edge connector, so ZX-81 users will need an extender card not yet available from Seikosha, so try Eprom Services. From DRG or Prism.

Tandy CGP-115 is the cheapest colour printer; it is very slow but can do pleasant graphics with the appropriate software. From retail shops.

Microline 80 is supplied with roll paper holder as standard. It uses ordinary typewriter ribbons. It has only one type-style. From X-Data.

Brother EP44 is a portable typewriter with RS232 connection. It is not really suitable as a computer printer but as a typewriter it is amazing. From local typewriter shops.

Olivetti JP101 is a quiet ink-jet printer; it has slightly blurred print but is reasonably fast.

Epson RX80 is the standard by which other printers are measured. It is versatile with clear print and all the facilities you are liable to need.

Star Gemini-10X was caught by a recent price reduction for the Epson, which it mirrors. It may be discounted further. From London Computer Centre or Micro Peripherals.

Hardware World

Mannesmann Tally MT80 is a good-looking printer, also heavily discounted.

Canon PW1080A is very fast and has a good range of facilities, including programmable characters. It is slower in near letter-quality mode. From London Computer Centre or Micro Peripherals.

Dyneer DW16 is a business users' daisywheel printer. The code is compatible with Diablo printers. From X-Data.

British Olivetti Ltd, 86-88 Upper Richmond Road, London SW15 2UR.

Dean Electronics Ltd, Glendale Park, Fernbank Road, Ascot, Berkshire SL5 8JB.

DRG Business Systems Ltd, Peripheral Division, 29 Lynx Crescent, Winterstoke Road, Weston-super-Mare, Avon BS24 9DN.

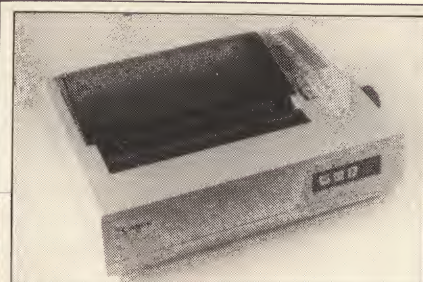
London Computer Centre, Grafton Way, London N1.

Mannesmann Tally Ltd, Molly Millar Lane, Wokingham, Berkshire RG11 2QT.

Micro Peripherals Ltd, Basing, Basingstoke, Hampshire RG24 0BY.

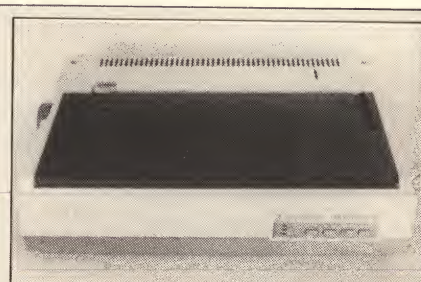
Prism Microproducts Ltd, 18-29 Mora Street, City Road, London EC1.

X-Data Ltd, 750-751 Deal Avenue, Slough Trading Estate, Slough, Berkshire SL1 4SH.



called he Seas: and G
Let the earth br
and the fruit tree y
in itself, upon the e
brought forth grass
the tree yielding fru
: and God saw that it
ng and the morning we
Let there be lights
de the day from the n
seasons, and for day
be for lights in the
on the earth: and it w
two great lights: the

Canon PW1080A



called he Seas: and G
Let the earth br
and the fruit tree y
in itself, upon the e
brought forth grass
the tree yielding fru
: and God saw that it
ng and the morning we
Let there be lights
de the day from the n
seasons, and for day
be for lights in the
n the earth: and it w
two great lights: the

Dyneer DW16

	Rec. retail price	Ports			Plan/thermal	Printing method	Characters per line	Dots per character	Characters per sec.	Time for test (secs.)	Max. paper width (inches)	Line feed button	Escape codes	Different type-styles	Graphics
		Centronics	RS232	Sinclair											
Alphacom 32	70			✓	T	Dot	32	8 × 8	—	50	4½	✓			✓
Seikosha GP50S	138			✓	P	Dot	32	7 × 7	40	152	5				✓
Tandy CGP-115	149	✓	✓		P	Pen	40/80	N/A	12	483 1440	4½	✓			✓
Microline µ80	229	✓	£68		P	Dot	80	9 × 7	80	170	8½				
Brother EP44	253		✓		T/P	Dot	80	24 × 18	16	383	8½	✓			
Olivetti JP101	265	✓	£40		P	Ink jet	80	7 × 7	80	67	9	✓	✓		✓
Epson RX80	286		£69		P	Dot	80	9 × 9	100	80	8½	✓	✓	✓	✓
Star Gemini 10X	286	✓	£80		P	Dot	80	9 × 9	120	69	—	✓	✓	✓	✓
Mannesmann Tally MT80	299	✓	£57		P	Dot	80	7 × 8	80	92	—	✓	✓		✓
Canon PW1080A	367	✓	—		P	Dot	80	11 × 9	160	48	10	✓	✓	✓	✓
Dyneer DW16	—	£378	£413		P	Daisy	101	N/A	14	330	10	✓	✓	✓	

EPROM SERVICES

QL, SPECTRUM AND ZX81 HARDWARE

SPECTRUM/ZX81 DATA ACQUISITION/CONTROL CARDS QL ADAPTOR

PRINTER ADAPTORS

RS 232 to centronics adaptor for use with Interface 1 or QL. Complete with cables. Other micros — Please enquire.

£39.95

ANALOG TO DIGITAL CONVERSION

8 channel 8 bit from £23.95
4 channel 10 bit input £27.95
8 channel 12 bit input £57.95
3 channel 8 bit output £26.95
also latches multiplexers etc.

ROM BOARD £14.95

Accents 2 × 276A Eproms or RAM to replace the spectrum ROM. Also contains 4 user definable key switches.

INPUT/OUTPUT CARDS

24 line MK3 £14.95
24 line MK1 + LED's £16.95
I/O accessories include output latches input multiplexers A/D, D/A and LED indicators

EPROM PROGRAMMER £54.95

A card for Z × 81 or Spectrum WYEL 2716/32/32A/64/128 complete with programming software requires external 27V-40V ie 3 × 9V batteries or PSU £9.95

COMMUNICATIONS

Single bidirectional RS232 £25.95
Dual didirectional RS232 £34.95
Dual bidirectional centronics £26.95
complete with centronics printer driver software

EPROM CARDS and CARTRIDGES

A range of board cards interaces and cartridges for ZX81 and Spectrum eg ZX81 card £18.95 also range of software in EPROM

TIME

Counter/timer — 4 × 8bit channels £13.95
*Interval timer — 3 × 16 bit channels £18.95
*Real time clock £24.95
*Event timer £21.95
*Crystal controlled cards

EXTENSION BOARDS

3 slot mother boards for ZX81 £15.95
Spectrum £16.95
complete with voltage regulation and power on indicator

CONNECTORS 23 and 28 way

Edge cards .75p
Edge connectors £2.50
Gold edge plugs £2.50
28 way IDC connector £4.95
Many other assembled cables & parts

EPROM SERVICES

3 Wedgewood Drive, Leeds LS8 1EF (0532) 667183
Large SAE for details. Export and trade welcome price include UK postage—overseas please add as appropriate
Industrial projects undertaken—please phone for details

POOLSWINNER

THE ULTIMATE POOLS PREDICTION PROGRAM

- **MASSIVE DATABASE** Poolswinner is a sophisticated Pools prediction aid. It comes complete with the largest database available — 22000 matches over 10 years. The database updates automatically as results come in.
- **PREDICTS** Not just SCOREDRAWS, but AWAYS, HOMES and NO SCORES.
- **SUCCESSFUL** SELEC guarantee that Poolswinner performs significantly better than chance.
- **ADAPTABLE** Probabilities are given on every fixture — choose as many selections as you need for your bet. The precise prediction formula can be set by the user — you can develop and test your own unique method.
- **SIMPLE DATA ENTRY** All English and Scottish team names are in the program. Simply type in the reference numbers from the screen. Or use FIXGEN to produce fixture list automatically (see below).
- **DISC/MICRODRIVE COMPATIBLE** All versions (except Apple and IBM) are supplied on tape, with simple instructions for conversion to disc/microdrive operation. (This seasons results are supplied with the package so that predictions can start immediately.) AVAILABLE FOR Spectrum (48K), Commodore 64, VIC 20 (+16K), AMSTRAD CPC 464, BBC B, Atari (48K), ZX81 (16K), Dragon, Apple II, IBM pc

PRICE £15.00 (all inclusive)



FIXGEN 84/5 AT LAST: No more struggling for hours to get the fixture list into the computer. FIXGEN has been programmed with all English and Scottish fixtures for 1984/5. Simply type in the date, and the full fixture list is generated in seconds. Fully compatible with Poolswinner. **POOLSWINNER with FIXGEN £16.50 (all inclusive)** Fixgen alone £5.50



COURSEWINNER v3 Coursewinner is designed to allow you to develop and test your own unique winning system. Using information from daily newspapers or 'Sporting Life', the most important factors can be input and analysed. The program is supplied with a database detailing best trainers and jockeys, and effect of the draw for all British courses. (Flat & National Hunt.) AVAILABLE FOR Spectrum (48K), Commodore 64, BBC (B), Dragon, Atari (48K), Apple II **PRICE £12.50 (all inclusive)**



phone 24 hrs
37 COUNCILLOR LANE, CHEADLE, CHESHIRE. ☎ 061-428 7425



SPECTRUM PROGRAMMABLE JOYSTICK INTERFACE

— PLUS SOUND

(Normally a £10 Add on)

from **Rainbow Electronics**

With or without THROUGH PORT — allowing connections of other add-ons to rear of interface

- PROGRAMMABLE** — Allows any joystick position to represent any key — without wires, or leads, or tapes!
- AMPLIFIED SOUND** — Internal micro-chip amplifier boosts Spectrum's beep with simple connection to ear or mic socket.
- EASY TO USE** — All joystick positions are programmed simply by selecting joystick position, pushing desired key on spectrum, releasing joystick, and then the key.
- ADAPTABLE** — Accepts diagonal positions.

ONLY £24
Extra for Through Port £3
plus £1 p&p.



Send Cheque or P.O. to: **Rainbow Electronics (YS)**
Glebe House South Leigh Witney Oxfordshire OX8 6XJ
Tel. Witney (0993) 5432

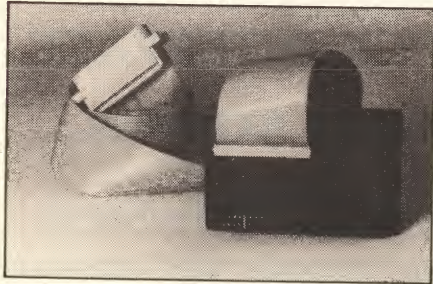
THE last year has brought about a vast change in the sort of add-ons available for the Sinclair machines. Many are technically brilliant while some would have made Heath Robinson blush. There are few add-ons still being produced for the ZX-81, with even fewer new ones, but there has been an avalanche of Spectrum designs, and the start of what promises to be an interesting range for the QL.

The most popular areas of interest — printers, joysticks and keyboards — are dealt with in separate sections but there have been some very innovative designs produced in other areas.

Kempston interface

Now that the Sinclair ZX printer has been discontinued, interfaces for full-size printers are becoming sought-after. One of the best of these for the Spectrum is the Kempston Electronics model E. The interface contains an EPROM that enables it, on power up, to direct the commands LPRINT and LLIST direct to the printer without the need for additional software. Also built into the EPROM are routines that allow the use of COPY for the Epsom and Seiko-sha range of printers.

It is housed in the standard Kempston case, plugs into the user port of the Spectrum, and is supplied complete with a cable to connect to the printer. The interface is dead-ended — that is it



does not have a through port for other add-ons — and that may be a problem if you want to use it at the same time as the Kempston joystick interface or any other dead-ended device.

Using the interface is very simple. If text only is to be printed then no setting up is needed, apart from POKEing an address with the number of columns required if that is other than the default setting of 80. That address is one of the unused ones in the system variables area.

To use the COPY command the interface must be set up for your type of printer. If you enter the direct command COPY:REM? the interface displays a menu page which shows its current status. You can then set it up for the Epsom range, Seikosha 100/250

printers or add your own routines.

Once you have set it up in this way any user-defined characters or graphic characters that appear in a listing will be printed as shown on the screen. For computer artists there is an enlarged setting which produces a double size COPY. Other settings are available to control the tokens, escape characters, and automatic linefeeds. The interface is by no means cheap but does contain many useful features.

Centronics interface

For QL users the Centronics interface from Miracle Systems is worth consideration. It plugs into one of the two RS232 ports on the QL and, using the default settings of 9600 baud, Parity off, can drive any standard Centronics printer. If you are still waiting for your QL but have an Interface One then Miracle Systems has another interface that will fit its RS232 port and which can be upgraded with an additional connector at a later date.

Beta Disk interface

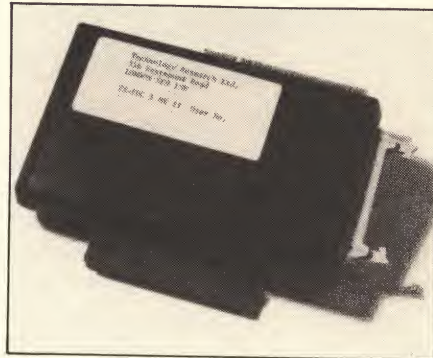
With Interface One now widely available and users turning their attention to fast storage, floppy disc systems are now available for the Spectrum. The Beta Disk interface from Technology Research has been improved continually throughout the year and now, in its final form, is the best available. It can handle up to four drives of either 40 or 80 track, single or double density format, in any mixture. The connector to the drive has the same format as that of the BBC drives and those can be used with it.

With the interface fitted it is useful to think of the Spectrum as having two distinct states. Either it is using Basic, from whence the disc commands can be called, or it is using the Disc Operating System (DOS); the only difference between the two is the call needed to bring the system into operation. As it uses the standard Sinclair syntax for LOADING and SAVEing Basic, code and data all you have to do is call the DOS prior to the command, which can be done from either Basic or machine code.

On power up the Spectrum runs under the DOS and requests you to input a password; that must be the same as that on the disc you are using. Once that has been done you then have access to the disc commands. As well as the LOAD and SAVE mentioned, which in addition to the normal commands allow you to auto-run CODE, you can CAT any of the drives, MERGE Basic programs, ERASE a file or MOVE which

moves up all the files to fill any spaces left on the disc by ERASE. Random access to the disc is by PEEK and POKE.

With all that it only uses 128 bytes of memory which it reserves by moving up the BASIC program area. Normally it is completely transparent to the user and



there are very few programs, commercial or otherwise that cannot be used with it, unlike the microdrive.

With a 40 track, single sided disc drive the interface gives 97.5K of file space, the other 2.5K being taken up by the directory. The 80 track version give you 390K. The interface is expensive, but it does allow the use of drives that are not dedicated to one machine and you can use easily obtainable discs which cost less than £2.00 each.

Tape controller

If however you are content with using cassettes then the Tape Controller from Ness MicroSystems might be a worthwhile addition to your system. It gives you the ability to control an external device, such as a tape cassette deck, and also provides an inbuilt BEEP amplifier.

The controller is housed in a small box which sits alongside the computer. Leads connect to both the EAR and MIC sockets of the tape deck and Spectrum. Another lead connects to the Spectrum power socket, and two further leads connect to the remote sockets on two tape decks. If you do not want to use two separate decks for LOADING and SAVEing a connector is provided to join both leads to one deck.

On the front of the controller are two switches. One selects whether the unit is to be used as a tape controller or as an amplifier. The other has three positions; up and down to load and save programs, without requiring the removal of leads as is usual, and in the middle position the cassette motor of the relevant deck can be controlled by detecting a preset BEEP frequency. Also provided is a volume control and two LEDs that show how the unit is set

up. The instructions that come with the controller are very thorough and include a short machine code routine to allow you to save a program without pressing a key.

That is particularly useful in a filing program as you can process data from one deck to another without the need for human intervention.

Telesound amplifier

A simpler BEEP amplifier is the Telesound from Compusound. As with previous models that a small printed circuit board which fits inside the Spectrum case and outputs sound through the TV. It has, however, been modified this year to work on the troublesome Issue 3 and 4 Spectrums.

Fitting the unit is fairly straightforward, provided that you actually have the courage to open up your Spectrum, and should present no difficulties. Three wires have to be connected to the Spectrum PCB and detailed instructions are given with the unit. That is a fiddly procedure, particularly the connection to the speaker. Once fitted the unit can be stuck down and the case screwed back together. In some cases the unit may have to be tuned to get the best sound and vision on your TV and, depending on the make of your set may slightly degrade the picture. If the picture is not good to begin with then that degradation probably will not be noticed.

Software protection

Two notable utility devices have appeared this year. The first of those, the ROM SP from Camel Products allows you to retrieve programs stored in your own EPROMs. Up to 16K of EPROM can be used and there is the



option of either having the program load on power up or by pressing a push button. Camel also produces the PROMER-SP which allows you to SAVE programs and data into an EPROM.

The second device is the Software Protection Unit from Microbyte, designed to stop the computer crashing in

the event of either mains noise or a power failure.

The Spectrum power supply is plugged into the unit and a lead is then plugged into the power socket of the Spectrum. Inside the unit are six 'AA'



type rechargeable batteries and, if there is a power failure, the batteries take over to keep the Spectrum working. In normal use the batteries are kept topped up and should last for up to 15 minutes in the event of a power failure. That will give you time to save the program if you have a battery powered cassette player to hand. The unit also contains a small circuit to cut mains interference. ZX-81 versions are available.

Graphics pad

The Format 4 kit from Orion Software is undoubtedly the most comprehensive and certainly one of the best graphic pads available.

Housed in a smart, black, ring binder the kit consists of two main parts. On the rings are 80 sheets of heavy duty paper, in four groups of 20. In the first group the sheets contain six large grids, each made up of four 8x8 grids and marked from 128 to 1 for easy calculation. Along the bottom are 15 smaller 8x8 grids for use as a scratch pad. The second group of sheets show a normal screen of 22 lines by 32 columns with each character square subdivided into an 8x8 grid. As well as the lines and columns being marked the plot positions are also shown. The third section shows an enlarged 1/4 screen, 16x11 character squares, subdivided into 8x8 grids. The last section is a straight forward 32x22 screen grid.

As well as that there is a clear plastic wallet to hold loose bits of paper, a plastic ruler marked with a pixel scale and a translucent screen for placing over pictures with a full 32x24 grid, each square subdivided into an 8x8 grid with markings around the sides showing the lines and columns, pixel positions and the memory addresses of each pixel line in hexadecimal.

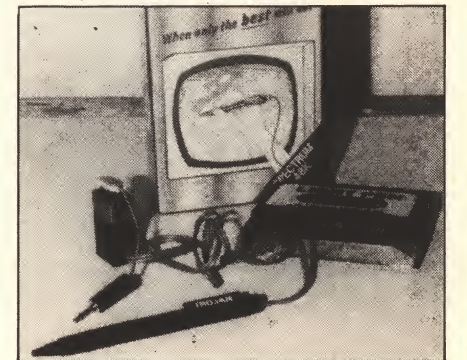
Inside the front cover are a note pad, seven coloured pens to use on the var-

ious sheets, a water soluble ink pen for use on the translucent screen, a calculator to help calculate the UDGs and lastly a C15 cassette with an optional drawing program. If you are looking for a serious graphics aid for your Spectrum then the kit should be near the top of your list.

Trojan light pen

If you prefer a more immediate means of putting graphics on the screen then the light pen from Trojan Products may be the answer. The pen is supplied with driving software on tape which allows you to draw pictures or to choose from a menu.

Once the software has loaded the pen simply plugs into the EAR socket of the Spectrum, it takes its power from a PP3 battery that is also supplied. On screen



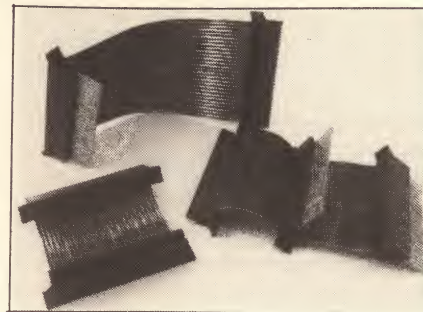
you are given the option of drawing or instructions. The instructions show how to use the pen and also how to modify it to use in your own programs as a menu chooser. All that needs is about 20 lines of Basic and 200 bytes of machine code, both of which are given. When using it in that way the code gives a number that corresponds to the line number to which the pen is pointing. In that way it is easy to pick from a list of options.

The drawing program has two movable points and by using these you can draw lines, circles and boxes, fill areas with colour or even draw freehand. Other options allow you to load and save pictures, change colours, insert characters or copy the screen to a Sinclair printer. The pen works well and is easy to use, overall a well thought out product.

Edge connector

Many users have written to Sinclair User about similar effects to that of the infamous ZX-81 RAM Pack wobble. The answer is the flexible edge connector from Classified Product Services. That is two Spectrum-style connectors joined by a piece of ribbon cable so that any wobble will not make an add-on

lose contact with the Spectrum. CPS also supplies an extender with an additional connector in the middle so that



two dead-ended add-ons can be used at the same time, such as printer and joystick interfaces.



The ZX-81 continues to sell well and the occasional add-on is designed for it. The Universal Modem Adapter and the Azmic system from Comprosys Ltd, being just two. The Modem adapter is one of the cleverest designs for the '81 for a long time. It allows it, with a modem, to access Prestel and the like. No mean feat as Prestel uses both colour and 40 characters to a line. Not only can the adapter handle that but it can also communicate at all the different rates required by databases, enhance the cassette SAVEing and LOADing and COPY Prestel screens to the ZX printer.

The Azmic System by the same company offers far more facilities than could be listed here. Its features include, however, a hi-res bit mapped screen, connection to any full size printer using either RS232 or Centronics, an assembler/monitor and extra keywords.

At the other end of the Sinclair scale many new products have been announced for the QL, and, once the details of the QDOS become more readily available, we can expect the level of hardware support for it to equal, if not exceed that of the Spectrum and ZX-81.

Kempston Printer interface Model E, £55.00 inc. from Kempston Micro Electronics Ltd, Unit 30, Singer Way, Woburn Road Industri-

al Estate, Kempston, Bedford MK42 7AF. Tel: (0234) 856633.

Centronics interface, £49.00; £5.00 for additional connector. Miracle Systems, 6 Armitage Way, Kings Hedges, Cambridge CM4 2UE.

Beta Disk interface, £85.00, exclusive of VAT and postage, from Technology Research Ltd, Unit 18, Central Trading Estate, Staines, Middlesex TW18 4XE.

Tape Controller, £19.95, or £16.95 in kit form, plus £1.50 p&p, from Ness Micro Systems, 100 Drakes Avenue, Inverness IV2 3SD.

Telesound beep amplifier, £9.95 inc. from Compusound, 32-33 Langley Close, Redditch, Worcestershire B98 0ET.

ROM SP, PROMER-SP, both priced £29.95 exclusive of VAT, from Carmel Products, 1 Milton Road, Cambridge CB4 1UY. Tel: (0223) 314814. EPROM erases also available from £18.95 plus VAT.

Software Protection Unit, £13.95 inc. from Microbyte, 19 Worcester Close, Lichfield, Staffordshire.

Format 4 graphics pad, £16.95 or £21.50 with the drawing program, plus £2.50 p&p, from Orion Software Products, Pippbrook Mill, London Road, Dorking, Surrey.

Light Pen, £17.25 inc. VAT and p&p, from Trojan Products, 166 Derlwyn, Dunvant, Swansea SA2 7PF.

Flexible edge connector, £8.50 (£12.50 for three way extender), from Classified Product Services Ltd, Shire Hall, The Sands, Appleby-in-Westmorland, Cumbria CA16 6XN. Tel: (0930) 52204

Universal Modern Adaptor and Azmic system from Comprosys Ltd, 29 Campden Road, South Croydon, Surrey SR2 7ER.

campbell systems for spectrum 48k

Can YOUR Database Handle This?

With MYRMIDON'S Micro-Print © up to 51 columns!

PROFESSIONAL FILE MANAGEMENT, DATA RETRIEVAL AND PRESENTATION ANY ADDRESS LISTS, INVENTORY, CUSTOMER OR PERSONNEL RECORDS...

MASTERFILE can!

Microdrive commands included; 32, 42 or 51 characters per line!; 26 fields per record; Unrestricted number of records; 36 user defined Display/Print formats; Fast search & sort facilities; Around 32K of RAM available for data!

Now with **MF-PRINT** and MASTERFILE version 09, you can format your data for a full width printer!

Works with most popular printer interfaces. Fully variable report widths (over 100 columns) and lengths. Powerful numeric editing and column totals. Almost no reduction in space available for data.

Many more products available. Send SAE for details!

MASTERFILE version 09 £15.00

MF-PRINT £ 6.95

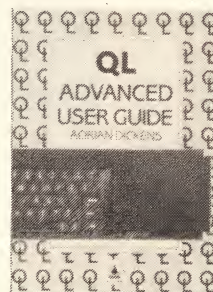
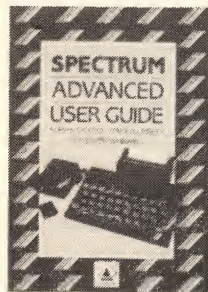
MASTERFILE with MF-PRINT £19.95

All programs mailed 1st class by return. Prices include VAT and postage within Europe. New Address: Campbell Systems (Dept. AN) 57 Trap's Hill, Loughton, Essex. IG10 1TD. England 01-508 5058



ADDER

for Advanced Computer Books & Software



*SPECTRUM ADVANCED USER GUIDE—by A C Dickens, M D Plumley and L Whewell, this book is an essential supplement to the manuals provided with the components of the ZX Spectrum systems. Both software and hardware aspects of the Spectrum system are covered including: a complete BASIC reference section, detailed screen maps, microdrives, Interfaces 1 and 2, the network, ROM modules, joysticks, a full Z80 reference section, detailed ROM routines with many practical examples of their usage—in short, every Spectrum owner's dream come true! ISBN 0 947929 02 9, 232 pages, £7.95

*QL ADVANCED USER GUIDE—by A C Dickens, this book is the authoritative guide to the Sinclair QL System. It provides a complete introduction to QDOS, covering multi-tasking, transient programs, resident procedures, heaps and stacks, traps and utilities, 68008 assembler programming plus much more. All of these features are illustrated by practical examples, and the powerful QDOS Experimentor program allows many facilities to be tried out from BASIC. All of the programs from the book are available on a microdrive cartridge which can be purchased with the book. ISBN 0 947929 00 2, 352 pages, £12.95 (book) £9.95 (+VAT) (microdrive cartridge)

Order from: Adder Publishing, PO Box 148, Cambridge CB1 2EQ
☐ Spectrum Advanced User Guide (book) £7.95 (No VAT)
☐ QL Advanced User Guide (book) £12.95 (No VAT)
☐ QL Advanced User Guide (microdrive) £11.44 (Incl. VAT)

Please add £1.50 p&p/book

I enclose my cheque/P.O. for £.....

NAME.....

ADDRESS.....

CITY..... POSTCODE.....

Dealer enquiries welcome

ALL PRICES CORRECT AT TIME OF GOING TO PRESS BUT SUBJECT TO CHANGE

SUPER-LEAGUE

For use on 48K SPECTRUM

Acclaimed the best football simulation game....

"...after using it for several days I had to write and congratulate you on a really excellent program." - A.M. (Middlesex)

"The program is addictive. Last night I spent from 11.00a.m. to 11.30p.m. taking Leeds to win the championship by 1 point." - I.B. (Shropshire)

"I must congratulate you on an outstanding game well worth every penny. Thank you." - S.S. (Essex)

"My son and grandson in Australia are delighted...a gem...neither of them have stopped playing it. Thanks for a great game." - N.G. (West Sussex)

"I thought I should write because of your excellent game." - Mr. McD. (W. Mide)

Note the following true-to-life features....

● Full 11-a-side match action

● On-screen commentary showing current score, name of player in possession, time played

● Scale 90-minute game

● Individual player skills

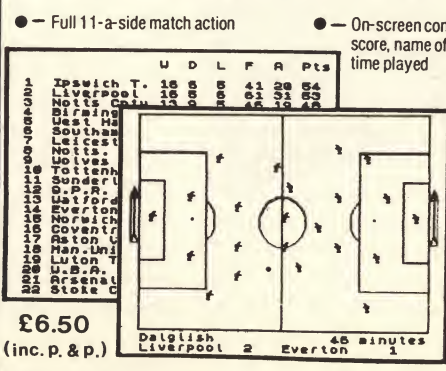
● Full 22-team League

● 42-match season

● League table and results on screen and printer

● Save season to date on tape

● Also form any league of 22 teams and pick your own players!

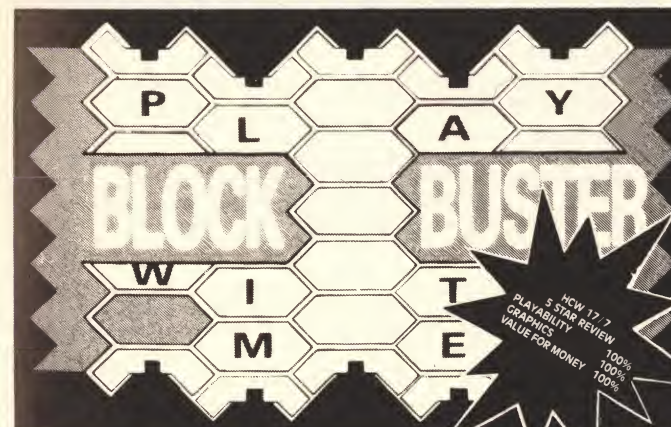


£6.50 (inc. p. & p.)

direct from
CROSS SOFTWARE
72 Swanshope,
Burney Drive,
Loughton, Essex, IG10 2NB

Also available from selected stores in the London area (including VIRGIN GAMES CENTRE, Oxford Street, W.1)

ADAPTED FOR 48 SPECTRUM FROM THE POPULAR T.V. SERIES!!!



BLOCK-BUSTER is an exciting quiz game for ONE or TWO players and will appeal to a wide range of ages and skills with nine levels of play against the clock. The game tests speed of recall involving anticipation of the question as it 'unfolds' across the screen and speed of response dependant on the level of play. BLOCK BUSTER is supplied with the FIRST ADDITIONAL QUESTION TAPE FREE to maintain the challenge of the game. Now available at good computer shops or direct with cheque/P.O. for £5.95 inclusive. ACCESS and VISA orders taken on our 24 Hour Hotline.

COMPUSOUND

DEPT SA, 32-33 LANGLEY CLOSE, REDDITCH, WORCS B98 0ET
 TELEPHONE (0527) 21429 (21439 24 Hr. HOTLINE)

NEW LOW PRICES

QL CENTRONICS PRINTER INTERFACE £34.00 INC.

Simply plugs into the 'ser I' port of the QL and any CENTRONICS compatible printer (e.g. Epson, Kaga, OKI, Seikosha, Juki, Canon, Mainsmann etc.). Requires no special software.

JOYSTICK ADAPTOR £5.90 INC.

Enables any Atari compatible joystick (e.g. Kempston, Quickshot 1/2 etc.) to be used with the QL joystick ports.

Joystick adaptors £3.00 each when ordered with the printer interface.

All products come complete with instructions and a 12 month guarantee.

Make cheques/P.O.'s payable to Sigma Research. All prices include V.A.T. and P & P within U.K.



Sigma Research,
 231 Coldhams Lane
 Cambridge CB1 3HV

QL is a registered trademark of Sinclair Research Ltd.

COLOUR MONITORS

MICROVITEC	PRICE INCL. VAT.
1431 — 14" STANDARD RES. (BBC etc.)	£215
1451 — 14" MEDIUM RES. (BBC etc.)	£330
1431MZ — 14" STANDARD RES (SPECTRUM/BBC)	£255
1431MQ/DQ — 14" MEDIUM RES (Q.L.)	£275

ADD £10 FOR DELIVERY



SUPERB MONITORS
FOR Q.L. AND SPECTRUM
 (1451MQ/DQ) (1431 MZ)

MONITOR/PRINTER PACKAGES FOR SINCLAIR Q.L.

Monitor	with Printer	Package Price INCL. VAT
MICROVITEC 1451MQ/DQ	— BROTHER HRS + MAINS ADAPTOR	£425
	— EPSON RX80F/T	£585
	— BROTHER HR15	£715
	— HONEYWELL S11 CQ	£699
	— EPSON FX80	£725
	— EPSON FX100	£899
☆ Epson printers supplied with RS232C interface fitted.		
☆ RS232CI/F board £33.45		
☆ Convert your Epson parallel printer to RS232C. NOVEX 12/1800		
MONOCHROME	— Brother HR15	£549

Prices include Monitor and Printer Leads
 Add £12 for carriage.

ZEAL MARKETING LIMITED
 VANGUARD TRADING ESTATE STORFORTH LANE CHESTERFIELD S40 2TZ
 TELEPHONE 0246 208555 TELEX 547697

AT LAST!
the MISSING link from
ARCHIVE to QUILL!

QUILMERG

QUILMERG will take a document prepared by QUILL as a skeleton, and automatically inserts fields SELECTed from an ARCHIVE database, printing a complete document for each ARCHIVE record. Ideal for mail shots, newsletters, reminder letters, standard contracts etc.

Price £12.50 plus 25p p&p

UTILITY PACK

PITCH ASSOCIATES UTILITY pack for the SINCLAIR QL, comprises of 12 functions, procedures and programs. Including a print formatter similar to conventional BASICS PRINT USING, data INPUT & VALIDATION, RAM & ROM HEX & ASCII dumps and microdrive FILE editor, HEX-DECIMAL conversion and HEX maths.

Price £10.00 plus 25p p&p

SPECIAL OFFER — QUILMERG and UTILITY
for only £17.50 (plus 25p p&p) when ordered together.

ALL PITCH ASSOCIATES software comes supplied with an UPGRADE voucher. When new releases are announced, send us a microdrive cartridge and the voucher for the latest version.

PITCH ASSOCIATES
 39 Rockleigh Avenue, Leigh-on-Sea, Essex SS9 1LA
 Telephone (0702) 73002

QL Utilities

FOUR programs on one Microdrive for the Sinclair QL.
 "Nothing very dramatic . . . could be quite useful" (PCW).

1 "Boot"

Displays DIRectory in columns, and allows LOAding of programs with single key-press. Without this, the directory will overflow the screen if you have more than about 18 files on cartridge; also "load mdv1-program" has 17 keypresses!

2 Q Mark"

Will allow you to FORMAT a cartridge repeatedly (as recommended by SINCLAIR). You only have to enter a name and number once.

3 "Q Back"

Makes back-up copies of cartridges. One key-press will copy all files onto an empty cartridge, or you may single-step and choose whether or not to copy each file. If receiving cartridge is not empty, you will be asked whether or not to over-write any namesake of a file about to be copied. You can copy from many cartridges onto one until it is full.

4 "Q Prune"

Will delete any file with a single key-press, so be careful with this one!

Instructions

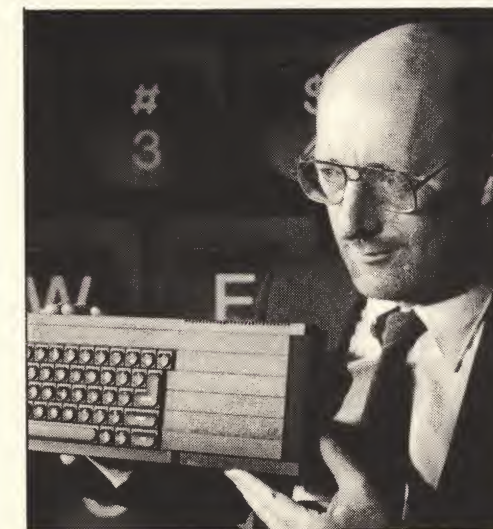
On Paper and in a QUILL file (for when you lose the paper!). All four programs will work with the extra microdrives if you get them, and have been successful with cartridges holding over 60 files each.

£10 from

WD SOFTWARE
 Hilltop, St Mary, Jersey, C.I.
 Tel: (0534) 81392.

Chapter 3

The QL Story



Once is an accident, twice is coincidence,
 and the third time is criminal action
 — or so the saying goes.

To the disappointment of the computer press
 and the public alike, deliveries of the QL
 were dogged by the same problems which
 delayed the ZX-81 and the Spectrum.
 Many felt that Sir Clive had over-played his hand
 once too often, and were unforgiving.

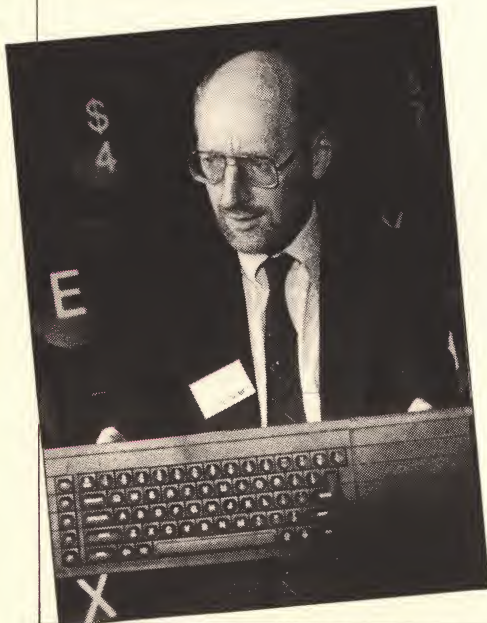
Sir Clive, for his part, felt the treatment
 handed out was uncalled-for and over-critical.
 John Gilbert relates the sorry story.



A huge stage filled the ballroom of the Intercontinental hotel, resembling a television set for Parkinson or Harty. It would soon be graced by the presence of a man who was just as well known in his own way, before an audience of journalists and computer industry figures gathered for what some would call the press event of the year, even though it was still only January.

Sir Clive Sinclair, chairman of Sinclair Research, introduced his new microcomputer by claiming it would be unlike anything else on the home and business computer markets. It would, he explained, make all other computers below the £2,000 mark almost obsolete and include hardware and software seldom seen on a machine under that price.

It was up to Nigel Searle, managing director of Sinclair Research, to detail the advances that Sinclair Research had made within the areas of computer design. It had taken more than a year to develop but the Quantum Leap was something special which everyone would want...



September 1982

The Spectrum was selling well and hundreds of companies had sprung up to support it. Sinclair had to admit that development for the peripherals, such as the Interface One, microdrives and Interface Two were not going well but with luck they would be available soon.

It was then that a new computer, eventually to be called the QL, was conceived. Sinclair anticipated the time when the popularity of the Spectrum would wane and other products would arrive on the market to replace it. Acorn Computers was close. Its BBC Micro was doing well in every way other than volume sales. The Acorn machine had taken the education market and put its stamp of authority upon it.

The imminent launch of the Acorn Electron would put Sinclair's only ace in danger. Nobody knew what it would be like but the advance specifications made it sound impressive.

Sinclair Research needed another computer and it had to be something so far in advance of its time few companies could catch up with its specifications quickly. As it would require a year to develop, it would have to be a machine which would last for several years and use technology which the company could claim would take it into the next decade. It also had to be a machine which could launch a new family of Sinclair microcomputers.

At the same time the machine had to have similarities with the Spectrum so that customers who were going to re-



main faithful to the Sinclair name would have no problems in exchanging one machine for the other. It needed technology that both home user and serious computer buff would find useful. The machine would also have to use the same type of price structuring scheme that Sinclair was famous for. The price would have to be phenomenally low.

A compromise chip

The specifications were a compromise to all those marketing and technological factors. In order to make the computer time-proof the new Motorola 68008 chip was chosen as the main processor. It was a 32/8 bit chip, much slower than the 68000 which had true 32 bit capacity, but it would run nearly

all of the software that the latter could handle. The saving in terms of money would be necessary to keep the retail price of the machine down when it finally went onto the market, a move then scheduled for late 1983.

The new Motorola chip would also allow Sinclair to implement a multi-tasking operating system on the machine which would be a breakthrough in the home computer market. Multi-tasking, a procedure in which several programs are run together, would allow the deficiencies in speed which might be criticised to be remedied by the software written for the machine.

As the 68008 was slow enough already and Sinclair wanted it to be the centre of the new multi-tasking operating system it was decided that a second

chip should take care of all the small jobs which, until then, had always been the province of the main processor. Those included the important job of making sure that information was taken in by the keyboard when keys were pressed and that the information so obtained was given to the correct devices. The secondary processor chip was to be the 8049 from Intel.

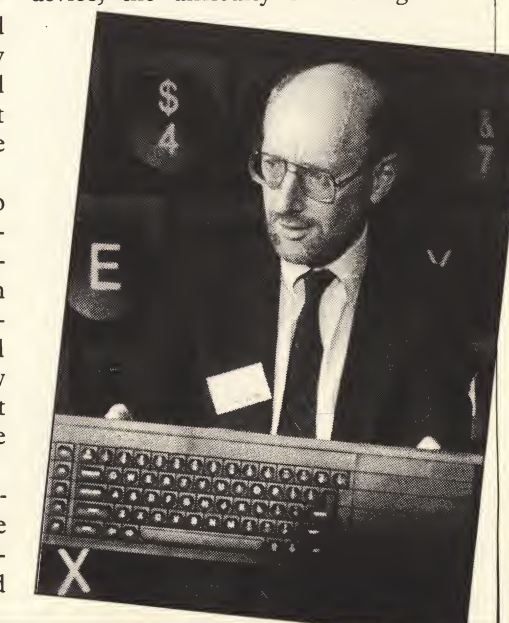
Communications

The machine had to be able to communicate with other Sinclair computers. For that reason the Sinclair RS-232C and network ports were included. The former would allow the user to link any RS-232 standard printer to the QL or to add joystick control in the unlikely event that games would be written for the machine. It would also allow communication between the QL and other computers if users had to transfer data from one machine to another.

Network was important also. It would allow users to link up QLs or Spectrums in the office or at school and to send data and programs around a chain of machines. The system would be ideal for small businesses where terminals which shared a common database of information were required.

The big risk

It was decided to provide twin microdrives with the basic QL system instead of the accepted double disc drive standard which so many machines, such as the Apple and Pet computers, had. It was certainly a risk. They were untested on the open market and few people had seen them. Coupled with the slowness of the ZX version of the mass storage device, the difficulty in making the





storage cartridges and the relatively small storage capacity on the tape loops meant the risk was big. But Sinclair was used to taking risks. The company had got away with such things before and it could always offer a hard disc interface later if the need arose.

As the QL was to use the 68008 the

most convenient internal memory size would be 128K. Most of it could be used by programs and data and approximately 32K could be used for a sufficiently complex screen display which would handle high resolution graphics and the window capabilities which would be built into the machine.

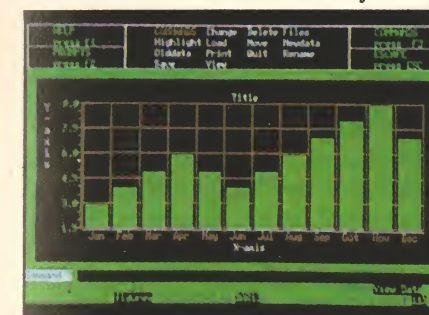
A peripheral slot would be built into the machine which would allow the user to add a cartridge containing a 0.5 megabyte card. That would give the QL approximately 620K of RAM, enough for even the most serious user. The slot could also be used for other devices such as ROM cartridges which required

more than the 32K which the planned ROM slot would provide and could cater for languages, new operating systems, graphics cards, and even a real time clock.

Looking good

Once the architecture of the machine had been agreed by the management of Sinclair Research the most important factor as far as marketing was concerned, the appearance of the machine had to be designed. The ZX-81 and Spectrum looked small and fairly elegant but both machines had the drawback of a keyboard which was anything but typewriter style or feel. Some critics had described using the Spectrum keyboard as similar to touching dead flesh.

The QL was to be an advanced machine so it had to look and feel like a portable, but it also had to have a keyboard which would stand up to wear and tear. Again a compromise was reached. A keyboard matrix similar to that of the Spectrum was used with a similar rubber underlay but the design was more advanced and the keys had a



positive feel to them when pressed.

The QL look was born...

Introducing QDOS

... Nigel Searle suitably impressed the journalists at the QL launch with his talk of the hardware facilities which, he claimed, would be available on the QL. But better things were to come.

He described the operating system as fully multi-tasking, allowing the user to run several programs at once. To put his point across he showed a TV demonstration.

It pictured a room, a Christmas tree in the corner. The door was open and snow was falling outside. Beside the tree was a picture on the wall. It was a miniature version of the Christmas scene and it too had snow falling.

Searle assured journalists that the picture could be reproduced and reduced, again and again, down to the minutest detail using the QL operating system called QDOS. It was that operat-



ing system and the language called SuperBasic which would prove the QL to be a master amongst machines.

Teething problems

When the hardware requirements of the machine were being dreamed up the Cambridge-based company GST was approached by Sinclair Research and asked to produce an operating system for the QL which was to be called QDOS — the pun was intentional.

Sinclair Research stipulated its requirements quite clearly and told GST that the operating system must fit into a 32K ROM to further lower the eventual retail price. GST went to work and within a couple of months put its proposals to Sinclair.

The QDOS system would be single user multi-task software with disc operating system, bit mapped graphics, a capacity for multiple screen windows and power to communicate with several programs at once using a pipe system

which was available in operating systems such as Unix.

Unfortunately, Sinclair Research had other ideas and, what was more, the operating system would not fit into the ROM space that the company had finally decided upon. Sinclair decided home-made was best and only three months before its intended launch decided to re-write the QL operating system using a team of inhouse programmers. That team was later to be immortalised when Sinclair used their initials, such as JB, AH, and JM, to mark the different versions of SuperBasic.

Psion software

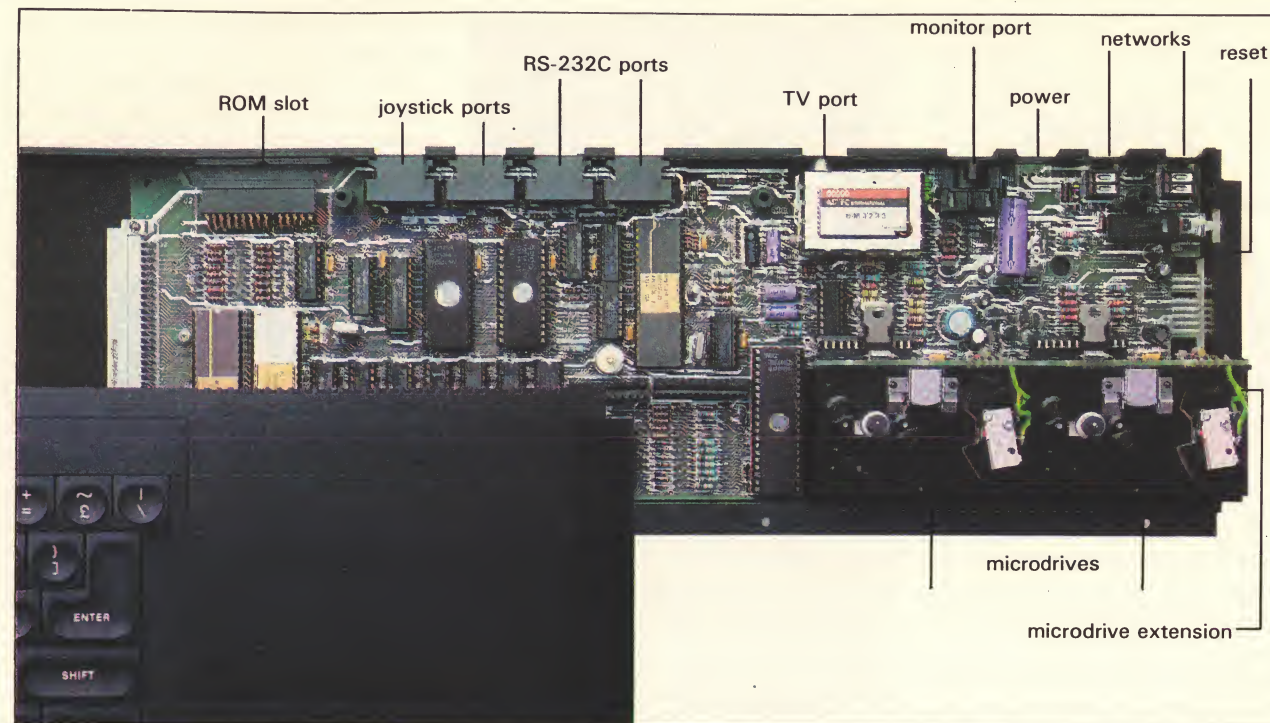
At the launch journalists were able to obtain provisional versions of the QL manual from the foyer and watch David Potter giving a demonstration of the four Psion software packages which were to be given away with the QL. The business packages, Abacus, Archive, Easel and Quill could be worked by even the most naive user.

The project leaders for the Psion software design were Charles Davis, technical director of the company, and Colly Myers, the leader of one of the Psion programming teams.

Each of the packages was produced on a VAX minicomputer for which Psion had written software which was able to emulate the QL of which then nothing has been seen. During nearly all of the development neither Potter, Davis or Myers saw a QL. When a machine eventually did arrive, with much secrecy, Psion hacked it to bits and it took more than a day to get the packages working with it.

All the bugs had to be ironed out. The problems were added to by the





continual stream of changes which were being made to both the hardware and software of the machine. It was not an easy task.

Déjà vu

After the press conference the journalists, somewhat numbed by the experience, went back to their typewriters. All had press packs which contained the



technical specifications and photographs of the new Sinclair machine and most thought that the company was about to release a machine which would put the computer industry into jump drive.

Everyone thought that the perennial Sinclair delivery problems were a thing of the past and that review machines would arrive, care of Securicor, within two weeks. Nigel Searle had said that the machines would be available from the beginning of February when the QL was to be advertised in most of the major computer magazines.

The advertising was to be on a massive scale with brochures pushed into the largest magazines — in size — and colour spreads in the smaller, popular, publications.

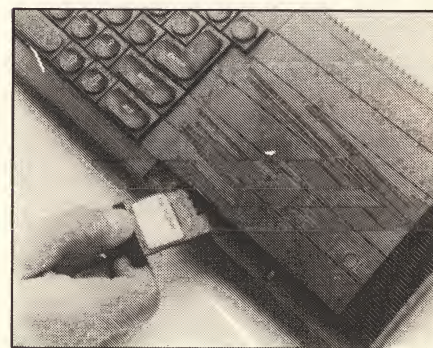
It did not take long for the press to recover from the honeymoon of the press launch and soon journalists were beginning to worry that their review machines had been lost in the post. Some began to wonder why the QL had been launched so early in the year.

To those who watched the market place and were wise to such things the answers were clear. Several computer companies were about to launch new models. Commodore was planning to tell the world about its new range of business micros and IBM had a few PCs up its sleeve. There was also some talk of Acorn launching a new business range.

Sinclair Research did not want the QL to be overshadowed by those other companies and so it dealt a double blow.

It arranged for a launch one week before IBM and also stated that its machine could outstrip everything under £2,000 in terms of performance. It had done that before but this time it decided to put its money where its mouth was and produce comparative bench tests for anyone who wanted them.

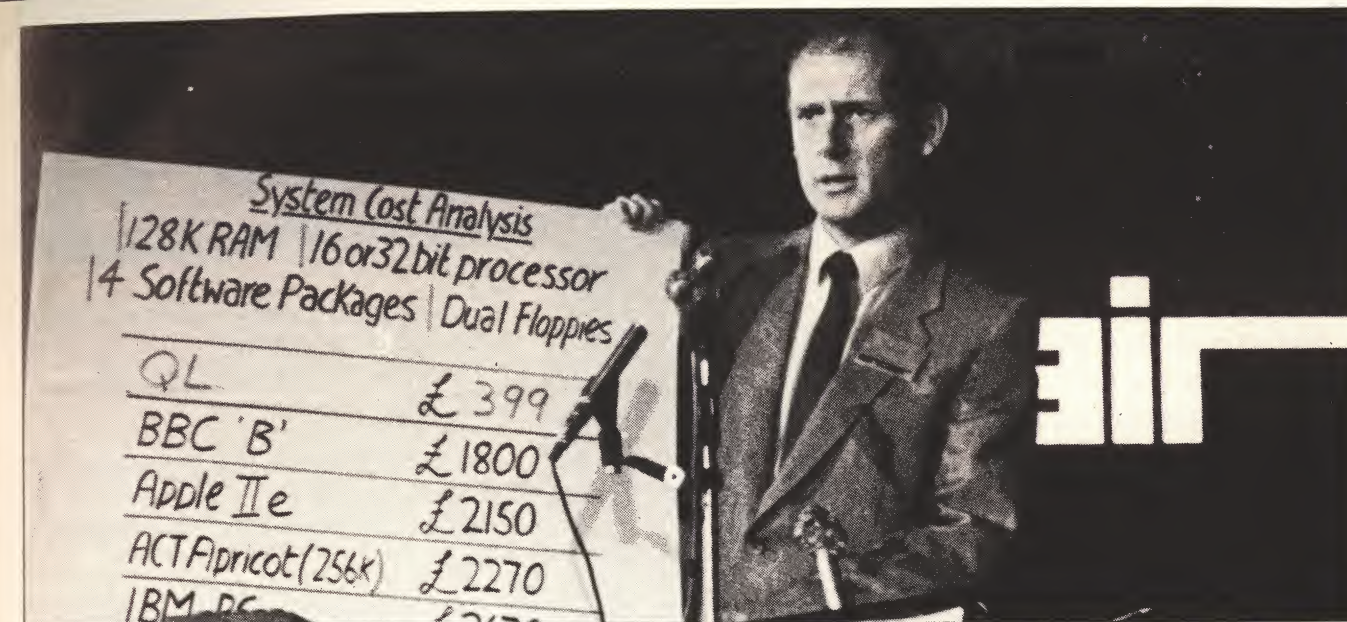
Few journalists took up the offer. They were too busy waiting for the postman, or Securicor, to turn up with



their QL. The machine was now over two months late.

Some editors could not wait. They had to have a QL and went to extraordinary lengths to say that their magazines had one. Many early reviews were of preproduction models, and had to be taken with a large pinch of salt.

The first reviews done on customers' machines appeared in *Popular Computing Weekly* and *Sinclair User*. Both magazines were later to revise those reviews when the new AH version of the machines followed that of the JB which had that familiar external EPROM sticking out the back.



The "dongle"

The first that the press knew about the EPROM — misnamed the 'dongle' — was when a selected few were invited to Cambridge in mid-April to see the machines, courtesy of Sinclair Research which was trying hard to scrape the egg off the laboratory carpet.

The head of the public relations department did a good job of explaining that the operating system would not fit into the planned 32K ROM so the designers were having to use a 48K ROM, part of which hung out of the now useless ROM socket. The good news was that turtle graphics commands and other little goodies were being added to fill some of the extra available space. The bad news was that the delivery of machines had yet again been delayed, and that customers fortunate enough to receive the first JB QLs would be unable to use the ROM socket until they returned their machines to Sinclair Research for an upgrade.

Another little problem that journalists were coming to terms with was that, despite what they had dreamed, the QL could not multi-task using SuperBASIC. Some believed that the original QDOS seen at the launch would multi-task SuperBASIC programs. At a press conference given later in July at the Carlton Towers hotel Nigel Searle could not understand how such a story got around. David Carlin, designer of the QL, simply asked "Why would you want to multi-task using SuperBASIC?" Another misconception about the power of the QL went out of the window.

Delivering the goods

By the end of the summer customers started to get QLs without the EPROM on the back. What was more, Sinclair Research was starting to deliver on time. Some people claimed they had received their machines within one week of an order though cynics saw that as an omen that production was outstripping demand.

Those customers were beginning to wonder what they could do with their machines. Two shows, the ZX Microfair at Alexandra Palace in early September and the Personal Computer World Show a few weeks later, provided the answer.

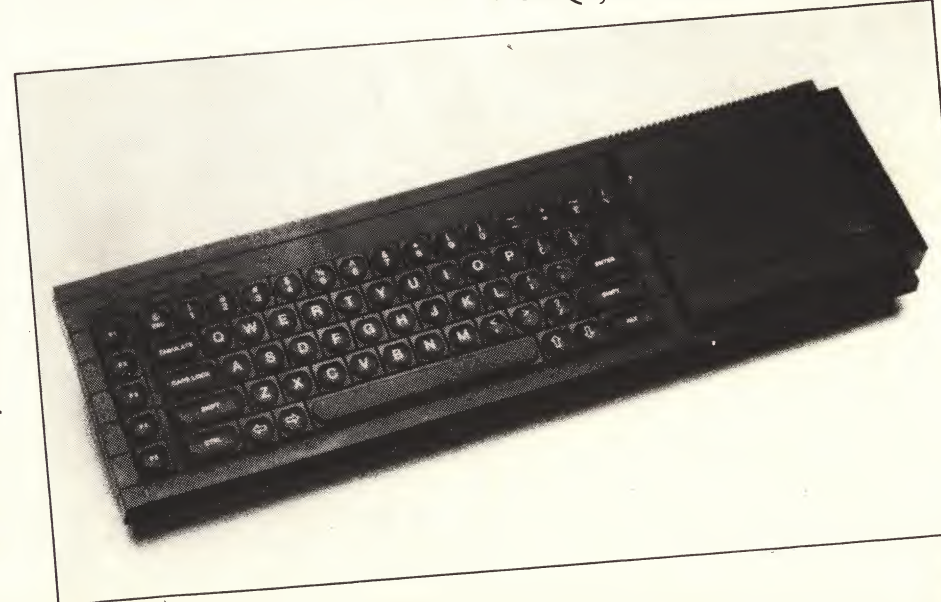
At the Microfair several companies showed off a range of products from the first books to arcade games. GST displayed its operating system, 68K/OS,

which was to have been the original QL operating system. It looked impressive and some people said that it outperformed QDOS.

Metacomco was at the show. The company had produced the first assembler package for the QL. It had also been one of the companies which tested the QDOS EPROM before it went into production.

Most important of all, Sinclair Research was at the show selling the QL. A few weeks later it was at the Personal Computer World show and announced that the machine was going into the retail sector of the market.

At the show Psion pointed the way to what the future of the QL software market could be like with its spectacular 3D chess program. Despite that nobody can tell what the future holds for the QL, not even Sir Clive Sinclair.



RELEASE THE TRUE POWER OF THE QL!

Write programs to unlock the multi-tasking power of your QL!

Metacomco now offers you three languages for the QL:

> Use **ASSEMBLER** for its speed, and for complete access to the QL's many features.

> Use **BCPL** for systems programming: writing games, utilities, and applications packages.

> Use **LISP** for manipulating data structures, and for exploring the world of artificial intelligence.

Each language comes as a complete development kit with a screen editor and documentation, for only £59.95.

THE ASSEMBLER

- > Standard Motorola 68000 mnemonics.
- > English error messages.
- > Produces code which can be EXECed, and run as a concurrent job.
- > External references allow linkage to high level languages and other assembler modules.
- > Macro expansions.
- > Position independent, absolute or relocatable code can be produced.
- > Conditional assembly.
- > Large range of directives.
- > Fully formatted listings.
- > 30 character variable names and 32-bit values.

BCPL

- > True compiler.
- > Multiple BCPL programs can run concurrently.
- > Full runtime library includes interfaces to QDOS graphics, window handling, file operations, etc.
- > Exception handling provides diagnostics for debugging.
- > Link loader allows linkage of separately compiled segments.
- > Assembler and BCPL modules can be linked together.
- > 32-bit variables use the full QL address space.

LISP

- > Full support of QL features including windows, graphics, and screen handling.
- > Compatible with Acornsoft Lisp for the BBC micro.
- > Interpreter.
- > Structure editor allows alteration of data structures.
- > Prettyprinter displays programs in structured format.
- > Tracer to aid in debugging.
- > Garbage collector automatically recovers spare memory space.
- > 28-bit integers and 250-character names.

THE SCREEN EDITOR

- > Full screen editor within a user defined window.
 - > Edits any ASCII file including SuperBASIC source.
 - > Comprehensive range of commands.
- The Development Kits are available now from HMV, Menzies and other leading retailers, or direct from Metacomco.

QL, QDOS, and SuperBASIC are the trade marks of Sinclair Research Limited.

METACOMCO

26 Portland Square, Bristol BS2 8RZ.
Tel: Bristol (0272) 428781

Phone today, or post this coupon to: Metacomco, 26, Portland Square, Bristol BS2 8RZ. Please send me:
 _____ copies of the QL Assembler Development Kit
 _____ copies of the QL BCPL Development Kit
 _____ copies of the QL Lisp Development Kit at £59.95 each.

I enclose a cheque for £ _____
 or please debit my ACCESS/VISA Account No. _____

Card expiry date _____

SIGNATURE _____

NAME _____

ADDRESS _____

POSTCODE _____

TEL NO _____

Price includes VAT, postage and packing UK mainland only. Delivery allow 28 days.

SUA

Chapter 4

Hit Squad



One trend in the software scene is the gradual recognition paid to the authors, rather than the publishers, of the most successful computer games. Some writers, such as Matthew Smith, have reluctantly attained superstar status. In this special interview Chris Bourne meets Mike Singleton — programmer of the 1984 Game of the Year, *Lords of Midnight*.

IT IS NIGHT. Save for a few who pick their way amongst the bodies of the dead, searching for the faces of friends and brothers, it is time for battle-weary warriors to rest. Wakeful, however, is Luxor, the Moonprince. The sounds of wailing and the shrill whistle of spears in the darkness assail his ears from all corners of the world. The reek of death and the stench of decay is forever present. And when, finally, dawn comes, there before the stronghold the serried ranks of the Foul stand, wave upon wave of black clad foemen, come out of the shadows of night to renew the attack.

What would you do? Take your stand there, and maybe die a glorious death, or flee with those men still fit to march, seeking new allies with whom to continue the long war against evil?

Such are the choices of Luxor the Moonprince, and yours too, if you play the adventure game **Lords of Midnight**. Not since **The Hobbit** have imagination and programming artistry been combined to such good effect in an adventure, creating one of the most engrossing and spectacular games of 1984. But however gripping the game, the true Lord of Midnight is not Luxor, nor his comrade Corleth the Fey, nor any of the great Lords whose citadels lie scattered about the land. He is an ex-English teacher from Wallasey, Mike Singleton.

Mike's first program had a distinctly seedy atmosphere about it, far indeed from the heroism and nobility of **Lords of Midnight**. "I first got into computing four years ago, with a horse-racing game written for the Commodore Pet," he says. No ordinary horse-racing game, however, since it was specifically designed for use in a betting shop. A colleague at school suggested the idea, knowing Mike's interest in mathematics, and Mike taught himself how to program in Basic. His first attempt was an accounting program, but he clearly had his sights on more exciting things.

"We went into association with a Liverpool firm of bookmakers, and produced a fully working version of the game," he says. But there was a problem with the gaming act, which draws a distinction between genuine races and entertainment. In order for bookmakers to operate, they are not allowed to entice customers into their shops with other forms of entertainment, and a test prosecution found against Mike's program.

To have one's first effort at professional programming squashed with

the force of the lawcourts might discourage some, but Mike persevered. His first programming success was with arcade games for the Commodore Pet, that grand old workhorse of the pre-Sinclair days in microcomputing. Through working for Petsoft, Mike came into contact with Sinclair, and wrote a set of games for the ZX-80, forerunner of the ZX-81.

"The deal fell through after I completed six 1K programs," he says, "but Clive was interested, and so was Sinclair Research." Travelling to Cambridge to visit the company he was shown a top-secret version of the ZX-81 with the operating system still in temporary form on an EPROM. "The

whole thing was still under wraps, and I was asked to convert my games for the machine." The result was **Gamespack 1**, a set of 1K programs all in Basic, and one of the very first commercial programs for the ZX-81. "It took two weeks work during the school holidays," says Mike, "and I got £6,000 in royalties. I don't think I've ever had better."

Mike Singleton has always been a games player. In his early teens he got hooked on board wargames, by companies such as Avalon Hill or SPI, whose complex rules and efforts at realistic simulation have had considerable influence on aspects of computer games. "It was wargames that started me off," he

says. "I invented my own board games, although none of them hit the market. I suppose I really came to computers from board games."

The hobby did not die with childhood. While pursuing a career in teaching, Mike was also participating in a play-by-mail game called **Starweb**, run by an American company, Flying Buffalo, creators of the *Tunnels and Trolls* fantasy role-playing game. There was a special game for overseas players. "We played by airmail, with one month between turns. It took me two years to win my game." Win it he did, and in late 1981 set up his own multi-player game, **Star Lord**, which was umpired by Mike with the aid of the Commodore Pet and a hard disk drive. There were about 700 players in the game whose moves were processed by the computer, and the multi-player game **Starnet** which runs on the Micronet network is directly based on **Star Lord**.

Enter at this stage one Terry Pratt, now the managing director of Beyond Software, but then editor of *Computer and Video Games* magazine. Terry approached Mike with a view to an article on Play-by-Mail games. The result was that Mike designed a game especially for the magazine, called **Seventh Empire**, which ran for about a year.

During that period Mike had also kept his hand in "churning out the odd arcade game" as he puts it. When Beyond Software was set up by the owners of *Computer and Video Games*, Terry Pratt moved from the magazine to run the company. "Terry started pressing me for programs," says Mike, "and eventually we came up with **Lords of Midnight**."

The idea was not a simple flash of inspiration, but the result of careful thought. "I was looking at the way things were going. I was interested in having a go at adventures, and it struck me that the graphics in most so-called graphic adventures could be improved. Pictures in ordinary adventures look very nice but don't tell you very much. I like to make everything very functional."

Certainly the graphics of **Lords of Midnight** are nothing if not functional. Every detail in the screen picture is of some significance to the game. In fact, in a real sense, the game is played through the pictures, which represent the view of the landscape from a particular character's position. Caverns, forests, mountains, towers, wolves and many other features all appear on the screen and affect the play of the game.

"The graphics were the initial thing," Mike explains. "I naively thought everything else would fall into place. I underestimated the mechanics of the game."

The game was specified beforehand in some detail, as the manual had to be sent off early to be printed, and a keyboard overlay was necessary because of the wide number of single-key entries required to play the game, where all decisions are taken simply by pressing the appropriate key. "That meant I was committed to producing a particular game without having confronted the actual problems of programming it. The major problem was the memory. With one or two K extra it could have been finished earlier. Believe it or not, it was streamlined three times. I squeezed the extra out of it by compression."

In fact, the game was a full six months in the writing. Incredibly, the whole thing was written on a single Spectrum without even the aid of microdrives. "The microdrives hadn't arrived," says Mike, laconically.

He is pleased with the final graphics routines, which give a possible 32,000 views on the screen from 4,000 locations. "I think the code is very neat," he says, which is probably a gross understatement. "It takes less than 1K to control, and a further 17K for the individual objects shown. Different-sized images are stored in advance, which is why the pictures come up so fast. It is awkward to have routines to expand or shrink objects, and when they get very small they look bad."

The fact that there is virtually no text in the game, and only single-key entry commands, is not simply a question of saving memory. "It would be inappropriate for the game to have text," says Mike. "There is such a lot going on the written commands would slow the game down."

If the graphics routines are unlike anything previously seen on the Spectrum, the game, the story and the construction of the Land of Midnight are also impressive.

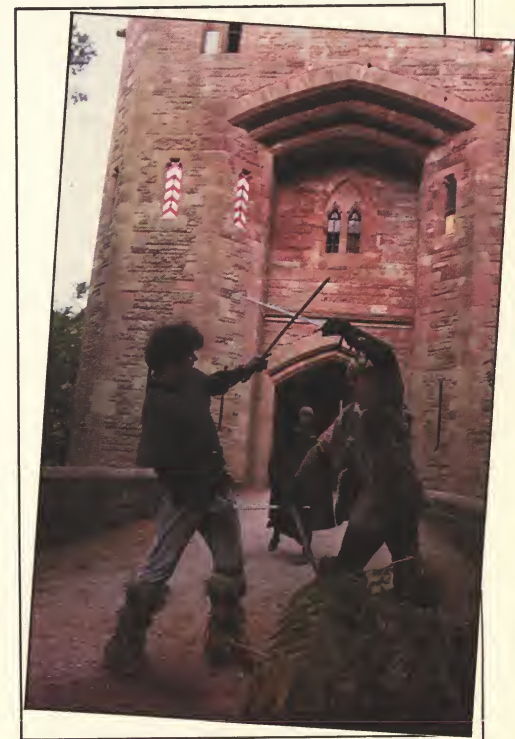
"It was entirely my creation," says Mike. "I started with the first chapter of the story setting out the characters of Luxor, Morkin and Doomdark." Luxor is the possessor of the moonstone with which he is able to communicate with his allies, thus enabling the player to control a wide number of characters. Morkin, his son, is the pure and fearless hero who must seek the ice crown, the source of the power of the evil Doomdark. Having established those major

characters Mike went on to draw the map, a rather more detailed version than the one provided with the game, including all the place names and major features. "I got the detail straight and then continued with the story. Rather than write the whole plot first, I let it run a bit and surprise me. I knew roughly where I was heading."

Tolkien, the guiding force behind modern fantasy writing, worked in a similar manner, not always knowing what was going to happen next in the story. Many of the elements in the game are found in Tolkien's work, the quest into the heart of the enemy stronghold and the fighting of an apparently hopeless war form the main strands of both the game and the *Lord of The Rings*.

"I am familiar with the literature, and of course it's impossible to avoid being influenced by Tolkien. I was after a game with a war in it, and the plot of the long defeat coupled with the quest for an object fitted in with the sort of game I was after. Once I had decided on the war/quest theme it was impossible to get away from Tolkien. There are other elements with a smack of Tolkien also. Fawkrin the Skullkrin is bound to be compared to Gollum."

Lords of Midnight has proved a great success, and by the time you read this the sequel, **Doomdark's Revenge**, will also have been released. "We always have a sequel in mind," says Mike. "But Terry wanted to wait before he committed himself to a second game. I've been working on it since finishing





Mike (centre) meets his doom?
Lords of Midnight."

Doomdark's Revenge is set in the Icemark, a land to the north-east of the area in which the earlier program is set. Shareth the heartstealer has captured Morkin and you must rescue him, with the aid of Rorthron the Wise, a character from the first game, and Tarithel the Fey, Morkin's lover, who appears for the first time.

The game is considerably more complex than Lords of Midnight, with 2,000 more locations and 128 characters — Lords of Midnight contained 32.

"I learned lessons programming Lords of Midnight," says Mike. "Towards the end I found ways of rationalizing the data structure to save space. It is a more complicated situation. Rather than having Good versus Evil, there are other factors, and more independent characters. The Land of Icemark is divided between five races. In the far north-east there are men under the command of Shareth, real baddies. Then there are giants, dwarfs, feys, and barbarians. All are independent and may fight each other."

The independence of the races means that Luxor must engage in complex diplomacy to win support in his quest. That may involve performing miniature quests for a particular object or goal desired by a race, thus providing possibilities for mini-adventures within the main game.

Another new development is the introduction of female characters, the evil Shareth and the good Tarithel. "I would like to have had female characters in Midnight, but left it to a later game." That was not out of raw sexism, but because of the limitations of memory. Involving both sexes means additions to the vocabulary of descriptions

and a gender factor to be added to each character, eating up the precious bytes. "There was a brief mention of a female character in the story; I was keeping the options open," explains Mike.

Although most of his time is taken up with work for Beyond, Mike Singleton retains his independence as a freelance programmer. He gave up teaching two years ago to work full-time on computers, and says he is satisfied with the switch. "I thought I would regret giving up teaching," he says, "but I don't. I do miss the kids — they kept me on my toes. It was always useful programming part-time, as I could take games in to the club at school. But it was getting to the stage where I had to make a choice because I was working all the hours God sent."

He is also sure that remaining freelance is the right decision, at least for the time being. "It's a nice idea to own your own company but it might restrict my programming time which would not be particularly appealing. The days of backroom companies are over. Perhaps if you had a set up with administrators you can trust..."

That attitude fits in with Mike's view of himself as a games designer rather than programmer, although his ability as a programmer is clearly in the front rank. "What would really suit me is to get a team of programmers and just feed them ideas. Terry had a position in mind for me at Beyond, but I think I make more money on the outside."

Mike plays fewer wargames now that he is writing them, but frequently has a game of Go, the Japanese strategy game which represents the next big challenge to programmers following the landmarks of draughts and then chess. "I play wargames every three months or so, and Go more frequently." He adds,

"I'm also in the pub a fair amount of the time."

He finds writing enjoyable, and welcomed the opportunity to write the story for Lords of Midnight booklet. "With anything like that there's feedback from the story to the program," he says. "At university I wrote a short novel — English teachers are always writing great works. It's called *The Eternal Empire* and there have been six or seven versions. When Terry mooted the idea of a novel as the prize for Lord of Midnight, I did say I wanted to have a go at writing it. But time was pressing."

When Lords of Midnight was released, a prize for the first winner was that the game should be written as a fantasy novel, using the winner's game moves as a basis. Whatever the book turns out to be like, we can at least say that literature's loss has been the game world's gain.



THE AGE OF THE R.A.T



CONVENTIONAL JOYSTICKS ARE DEAD!

The Cheetah Remote Action Transmitter is the most sophisticated computer controller available.

It has these features:

- Infra Red transmission — so there are no leads trailing across the living room. Just sit back in your chair up to 30 feet from your machine.
- Touch control — no moving parts, extremely fast, long life.
- No extra software required.
- Can be used with all Cheetah RAT/Kempston compatible software.
- Fits comfortably in your hand for long play periods.
- Comes complete with receiver/interface unit which simply plugs into the rear of your Spectrum.
- Compatible with all Sinclair/Cheetah peripherals via the rear edge connector.

Simply incredible at £29.95 including VAT and p&p.

Dealer enquiries welcome. Export orders at no extra cost.

Send cheque/p.o now to:

Cheetah Marketing Ltd. (Dept. AN), 24 Ray Street, London EC1R 3DJ. phone 01-833 4909

Cheetah products are also available from branches of

John Menzies WHSMITH Rumbelows

and all good computer shops.





"KNIGHT LORE" recommended retail price £9.95 inc VAT
Available from W.H.SMITHS, BOOTS, J.MENZIES, WOOLWORTHS
and all good software retail outlets. Also available from
ULTIMATE PLAY THE GAME, The Green, Ashby-de-la-Zouch, Leicestershire LE6 5JU
(P&P included) Tel: 0530 411485

Chapter 5

Software Scene



While some software houses are taking the Spectrum to its limits and beyond others doggedly continue to churn out ever more diabolical pieces of programmed junk. John Gilbert presents a personal pick of the bunch, and Chris Bourne takes an irreverent look at the dwindling ZX-81 software scene. Their talents are combined in listing the Top Ten Turkeys of 1984. Let the reader be warned.

Spectrum software

ACYNIC might argue that development within the software market in 1984 was almost non-existent.

The same type of game appeared as those which took the lead in 1983, the most popular being of the arcade variety. The programs were written in the same style and to please the same type of customers.

That is only a superficial view, however, and if you look at the games market as a whole, dividing it up into sectors such as strategy, arcade and adventure, you will see that substantial and sophisticated changes have taken place. Despite what some pundits have said you will find that the world of computer games is still buzzing with life.

Jet Set Willy

A number of cult games sprang up during the year. The biggest was the **Manic Miner** sequel, **Jet Set Willy**.

Manic Miner was one of last year's classic Spectrum games and programmer Matthew Smith brought Willy back again in an equally zany sequel.

Willy is living the decadent life after striking it rich down the mine but his housekeeper revolts after one party too many and will not let Willy go to bed until he has cleared all the mess.

You have to guide the debauched playboy round his enormous mansion full of lethal traps and a strange blend of domestic and diabolic mayhem.

It is worth the initial trouble. The graphics are clear, fast-moving, and varied, depicting ghosts, mad monks and animated toilet seats with equal smoothness and precision. Tactics need to be varied as well; only quick reactions will save you from the predatory security guards, whereas forethought is required to negotiate your way through the hazards of the giant Metatree or the unspeakable menace in the kitchens.

Sabre Wulf

Another cult game to appear on the shelves was **Sabre Wulf**. Produced by Ultimate it closely followed the format of the company's last best seller, **Atic Atac**.

The concept is similar to the earlier game. You must seek four parts of a hidden amulet to escape from the jungle maze; on the way you must battle a dazzling variety of foes, some weak and

others seemingly invincible. There are numerous treasures to inflate your score, some of which may prove of use in the game.

The game is played over an enormous maze of jungle paths and clearings.

One major feature is the various coloured orchids which, when eaten, have strange temporary effects on your abilities.

The disorientation orchid, which reverses all joystick controls, is particularly infuriating.

While the graphics are not entirely flicker-free, with the usual problems when figures pass over each other, they are up the high Ultimate standards and the action is fast.

Strangeloop

When **Strangeloop** came onto the market in the autumn Virgin Games confidently claimed it would beat games such as **Jet Set Willy** and those from Ultimate both in terms of popularity and sales.

The game, which involves a race around a robot factory to find the computer which has made everything go haywire, is well on the way to doing just that and may well turn into a 1985 successor to the Miner programs.

Another area of success for arcade software was that of the visually stunning game where the graphics mattered just as much as the plot. Into that category fall such offerings as **Death Chase** from Micromega, **Trashman**

from New Generation, and **Stop the Express**, a new addition to the Sinclair Research software library.

Stop the Express

In **Stop the Express** the evil Redmen have captured the ITA Express as it hurtles towards an unknown doom. As a special security agent you have to round up the Redmen before the train reaches the border, when they will be out of your jurisdiction and free.

There are two stages. In the first you have to run across the top of the train, jumping each carriage as you reach it. You must be careful, however, to avoid the attempts by the villains to eliminate you with their knives or fists. You should also beware of the low-flying electric pylons above your head.

Despite the near impossibility of your mission you have one friend, the Snakebird, which you can catch as you travel over the carriages and release into the arms of the Redmen, who will suffer the same fate they had planned for you.

Once you have completed stage one you will be able to get inside the train, where Redmen and other assorted nasties will lunge at you, trying to shorten your life.

If you reach the end of the train, the express will start to slow and you will be congratulated on your efficiency as an agent. Then you are promptly dumped on top of the train again to run the gauntlet with more Redmen in the next round.

Stop the Express is well on its way to

being a classic game. The idea is original and the graphics are of cartoon quality. It is also difficult enough to be addictive and, even when you have discovered the second level, you will want to keep playing.

Death Chase

Death Chase uses animated graphics to simulate the view from a motorcycle. The player moves on it through trees, chasing other riders and occasionally spotting tanks or helicopters. Speed and direction are both realistically simulated, so much so that the inevitable crash comes as a worrying shock.

Other riders cross the player's field of vision, weaving back and forth, but never out of sight for long. It is impossible to overtake those riders; they cannot fire at you, they cannot escape unless you crash. It is at those realistically-drawn characters that the player must shoot to kill. The player is practically immortal, with three lives per game, and an indefinite number of games to be played.

Trashman

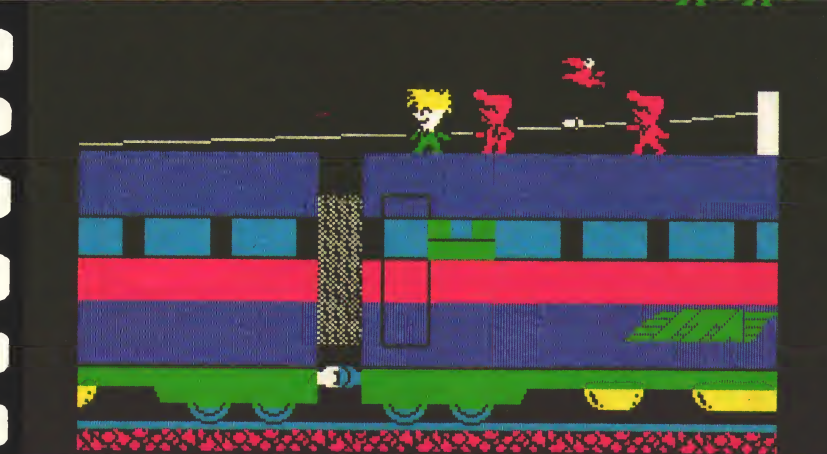
Trashman became well known for its 3D graphics and is a classic in its own right. The arcade game simulates a day in the life of a dustman as he collects the rubbish bins of suburbia and empties them into the dustcart. Unfortunately there is a time limit on each series of jobs, indicated by a decreasing bonus score. When it reaches zero, the residents complain about you and you get the sack.

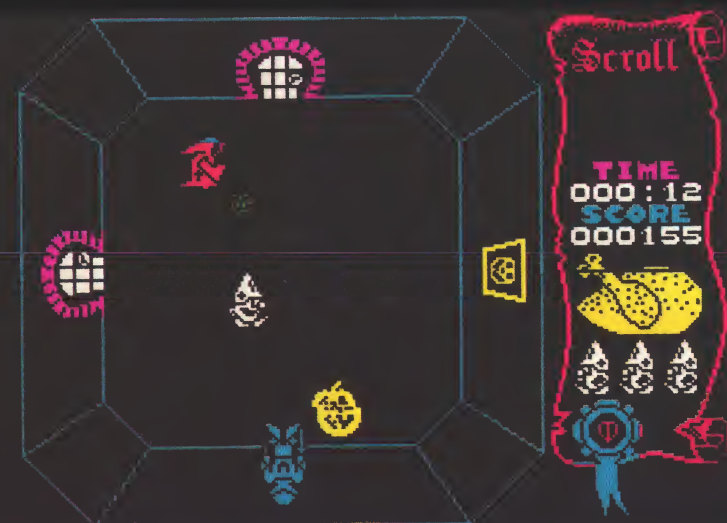
The screen displays a realistic view of small-town side streets with pretty red-brick houses in 3D perspective. When you reach the top of the screen the display moves to the next section of road.

The race against time would be difficult enough without all the extra hazards. If you walk on the grass you lose bonus points fast and there is traffic to avoid when you cross the road. The driver has a mind of his own and will not wait if you are slow, but drives on, forcing you to run after him. There are also vicious dogs which leap out to bite your ankles and slow you down.

The Alchemist

Following the successes in arcade games many software companies incorporated arcade movement into adventures. The movement started back in 1983 with the Carnell **Black Crystal** trilogy. Later Imagine, now no longer with us, released what some believed





This house is well kept and has two floors.

Also here is a Cartographer
The Cartographer says
"Gosh its cold."
Edgar says
"We'll never get to the
mountain at this rate!!"

then to be the ultimate in graphic adventures, **The Alchemist** for the 48K Spectrum.

You take the part of the alchemist who is searching for the parts of an ancient spell scroll which, when put together, will destroy an evil wizard. Your character can move around on the ground in human form or can transform into the guise of a giant bird and take to the air. All that can be performed with the keyboard or with many types of joystick.

To reach the parts of the scroll you must move past various evil monsters, including a colourful butterfly and a sheet-like ghost, which are all animated cleverly.

At first sight the game, with its cartoon maze and wizard as the central character, may remind you of the Ultimate Atic Atac. The extra features of the game, however, compensate for the similarity and there is much more variation in the maze construction and the creatures which will be seeking your blood.

Lords of Midnight

Graphic adventure of the year, however, was surely **Lords of Midnight** from Beyond. Like Atic Atac it soon gained cult status and created a new genre in adventure gaming. It is a truly epic adventure. You play the part of Prince Luxor and his allies, whose movements you control through the telepathic power of the moonstone. During the day you, and those you have won to your cause, travel the land raising armies, fighting battles, and sending ambassadors to recruit more Lords to your side to defeat the armies of Doom-dark.

The most striking feature is the superb graphics system, whereby the screen displays your view of the land from where you stand. Distant objects become bigger as you move towards them and you will need to develop a keen eye to spot significant features on the distant horizon.

The graphics are built of standard pictures for various parts of the landscape; it is claimed that there are 32,000 possible views, all different in some respect from each other. Fortunately there is sufficient variety and detail in the pictures to sustain interest and the information contained in them is vital to the game, as there is very little accompanying text.

Avalon

In the autumn of 1984 Hewson Con-

sultants brought out **Avalon**, another step forward in 3D graphics. The game casts you in the role of a mage who has to defeat an evil warlock. In order to do that the mage must negotiate the terrors of the warlock's underground caverns, going deep into the bowels of the earth. The mage must discover treasure, fight monsters and gain spells which will ultimately defeat the warlock.

The plot sounds, and is, mundane and is on the same level as other *Dungeons and Dragons* games or even Atic Atac. The graphics are, however, something special. All objects are displayed in 3D, the mage can rotate and float around the scenes and doors open outwards in perspective. Hewson Consultants called it an adventure movie when it was first launched. The graphics may not be that spectacular but they are very close.

Sherlock

Text-only adventures were not forgotten by software companies during the year in which Level 9 Computing converted its classic adventures to run on the Spectrum and produced one of its best programs, **Lords of Time**, which won several rave reviews.

The successor to the Melbourne House classic **The Hobbit** was released in September. **Sherlock** uses a format similar to *The Hobbit* though only few locations are shown graphically.

The game casts you in the role of the famous detective and with your faithful and infuriating side-kick Dr Watson, you must solve a series of mysteries in order to crack a murder case.

Characters within the adventure can be questioned, the evidence is there if you can find it and the atmosphere of Conan Doyle's Victorian world can be felt from the moment the game was loaded. It is bound to become as great a classic as *The Hobbit* which, incidentally, was written by the same team of programmers.

Ket Trilogy

Another all-text adventure quietly achieving popularity was the **Mountains of Ket** billed as the first of a trilogy of games by Incentive Software. The third and final part arrived in the fall of 1984 but the first still remains a firm favourite with adventure fans.

The Lords of Ket rule a strife-torn land where magic and mayhem are normal. Raiders from the east sweep in and devastate the countryside, spurred by the villainous priest-king Vran Verusbel, arch-mage of the cult of Mad

Monks, and by the beautiful though utterly evil priestess Delphia.

You know little of that as you languish on Death Row awaiting execution for a crime you did not commit. Suddenly, on the eve of your demise you are summoned before the Lords who give you a stark choice — die tomorrow or travel east through the perilous Mountains of Ket. Your mission is to destroy Vran and Delphia and thus end the troublesome raiding. Of course you accept but, in case you were thinking of running away, the Lords have placed a magic assassin bug called Edgar on your neck to ensure loyalty to the cause.

Your prowess, energy and luck determine your likelihood of surviving in combat and it is wisest to steer clear of fights wherever possible.

As in *The Hobbit*, your score is calculated on a percentage basis and there is a save-game facility. The creature interaction is somewhat limited, however, and talking to the characters can be difficult.

Bridgemaster

Few traditional mind games packages were produced but the pick of the bunch has to be **3D Strategy**, from Quicksilver, and **Bridgemaster**, from Bridgemaster, a company which has since wisely changed its name to Serin. **Bridgemaster**, for the 48K Spectrum, is nothing to do with the programs which claim to play full versions. It fulfils the worthier aim of teaching you how to play Bridge.

The course is split into a series of lessons using the Acot system, based on hands dealt by the computer. It starts simply by giving the user a chance to see play in action and you can follow play on the screen with an audio cassette which provides background narrative.

When the narrative is finished you have a chance to play the hands discussed in the lesson and the computer will inform you of any mistakes you make. There are 55 deals which you have to run through in that way, although the package makes each interesting in turn.

3D Strategy

3D Strategy is played in a three-dimensional cube but, apart from that complication, is not much different from noughts and crosses, although that one difference is a big one.

The computer will play two types of game with you. The tactical play will make the computer move in a risky and

decisive way with a good chance for the calm human player. The positional play option will make the computer move in a considered way. That means the human player can tailor the game to any style of play.

To make the competition even tougher you can put a time option, for minutes or seconds, into effect. It operates in a similar way to blitz chess, where the player and computer must complete a game before time expires.

Superchess III

Only one chess program of note was launched in 1984 and that was **Superchess III** from CP Software.

As well as allowing you to play games at 10 levels of difficulty, the program will also analyse chess problems of up to four mating moves. The option will also give you an idea of how mate in various positions should be achieved.

A relative beginner can play on level one and have the feeling that he is a good match for the computer. On the higher levels, which can take some time to play, the expert at chess will find the computer an impressive opponent.

All the usual features which have become part of computer chess games are included. There is a recommended move option, an option to change sides halfway through a game, and a routine which will allow you to change the colours of both the pieces and the board.

An extra feature which is not included in most other chess games for the Spectrum is the technical information option. The section will tell you briefly about the techniques which the computer uses to find its moves and which are used when it analyses problems.

Match Point

The most exciting strategy war game of the year was **Stonkers** from Imagine, and the best simulation, by a long chalk, was **Match Point**, a game from Psion which followed past successes **Flight Simulation** and **Chequered Flag**. Released on the Friday before Wimbledon Fortnight Match Point was an instant best seller.

The screen display shows Centre Court at Wimbledon. The panorama provides a view of the net, umpire, crowds moving their heads, and even the benches on which the players sit after a game.

You can play either a quarter-final or semi-final if you are inexperienced, or a final if you want to be thrashed soundly by the other player. If you have no

human friend with whom you can play, the computer will always be on hand for a game.

When you start to play you may have difficulty distinguishing between the ball and its shadow. Once you have played a few games, however, the effect falls into place and adds a three-dimensional quality to the game. The shadow of the ball is the only unrealistic feature of the display as, in real life, the players cast the shadows and not the ball.

Stonkers

In Stonkers you are given command of land-based forces, including tanks and infantry, which are grouped in the top right-hand corner of the screen. The enemy is based across the river on the other side of the screen and to reach a suitable combat position one or other side has to cross the bridge.

When you start to play, a map of the whole theatre of war is displayed. You can issue commands to active units on the map by positioning a cursor over the map position you require.

One of two actions will occur. If you are on the big map you will zoom in to a detailed display of the location specified. If you are already zoomed in you can position the cursor over a fighting unit and by moving the cursor to the position you want it to take up, it will start to move towards that locality.

Mugsy

Another strategy game that grew in notoriety was **Mugsy**, a land management game with a difference from Melbourne House. You play the part of a gangland leader. The game requires you to make decisions as to how much money to extort from your protection rackets, what to set aside for bribery and how much to spend on arms for your underlings. Each decision you make is portrayed in cartoon form, with the words in speech bubbles; a respectably large number of screens complement the story, in the style of old gangster movies. The pictures look to have been designed by a genuine artist and add immensely to the atmosphere of the game. At the end of each 'year' you also have an animated cartoon of an event during that year. There are only two of those short films but both are interesting to watch and they use the Spectrum graphics capability to the full.

Inevitably the graphics will tend to become uninteresting after they have been seen enough times but while they remain fresh they are a delight and make the game a pleasure to play.

This house is well kept and has two floors.

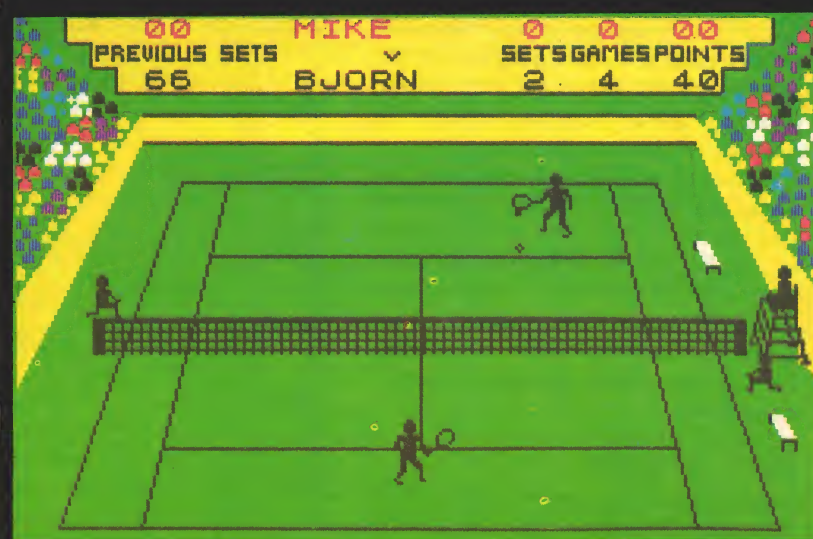
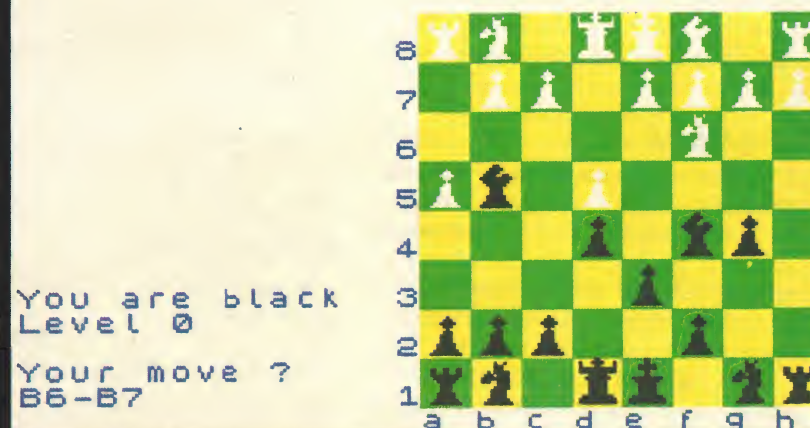
Also here is a Cartographer
The Cartographer says
"Gosh its cold."
Edgar says
"We'll never get to the mountain at this rate!!"

What shall we do? **SAY HELLO**
The Cartographer says
"Speak up sonny I'm a bit deaf"

What shall we do? **HELP**
Don't just kill everyone!

What shall we do?

CP Software Copyright (C) 1983
AUTHOR: Chris Whittington.



Supercode

Although their releases were not so noticeable as those of the game companies several software houses brought out utility programs which went a long way in helping Spectrum owners write their own programs. The ultimate in such

utilities was **Supercode** which contained routines for doing everything from scrolling the screen to producing sound effects, containing 102 machine code programming aids.

Many of the routines are useful in games writing, letting you scroll in eight directions in high- or low-resolu-

tion or the attributes. Other routines produce a number of sounds and there are also six ways of protecting a program, including ON ERROR, GOTO and ON BREAK GOTO.

All the usual toolkit of routines are available, block delete, remkill, compress and two kinds of re-number, plus other useful utilities such as hex-dec converters, hex loader, tape head reader and many more. There is even one which accents letters.

HURG

A new concept in software, the arcade games designer package, was also launched last year. The first onto the market was the Quicksilver **Games Designer**, released in 1983, and Melbourne House followed with **HURG** in the spring of 1984.

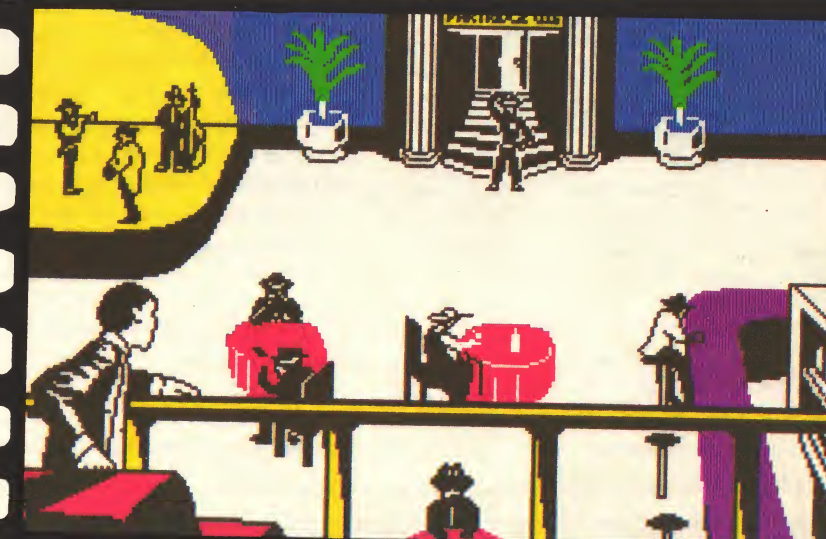
HURG is a competent program to help you design arcade games but it necessarily limits the scope you have. You will not be able to create a new version of Manic Miner or Atic Atac. You will not even be able to emulate the Melbourne House Penetrator. You will, however, after many hours of practice, be able to create your own version of Donkey Kong or Galaxians without too much difficulty.

There are several impressive features to the program. First, it is menu-driven, like the Quicksilver Games Designer. On loading, you are presented with a series of choices on the screen. Choosing 'Play Game' allows you to test whatever game on which you are working. Choosing 'Edit Game' will take you to a new menu with more choices, like writing a title page or specifying conditions for proceeding to a new stage of the game — e.g., when all the space invaders are destroyed.

What HURG does is eliminate much of the tedium of programming — it still takes a long time to create a game but nowhere as long as it would starting from scratch — but what it will never do by itself is give you the originality and inspiration of top games designers.

Although the software market has tended to stay with arcade games in 1984 the standard of those programs and the new ideas which have been generated prove there is life after **Space Invaders**. Looking through the games, however, one wonders whether the 16K Spectrum is worth buying any more as few software companies are producing anything for it.

Software houses did well in 1984 but there is room for improvement in all areas. Lets hope 1985 will see it.



Jet Set Willy, £5.95, Software Projects.
Sabre Wulf, £9.95, Ultimate.
Stop the Express, £5.95, Sinclair.
Death Chase, £6.95, Micromega.
Trashman, £5.95, New Generation.
Alchemist, £5.50, Imagine.
Atic Atac, £5.50, Ultimate.
Lords of Midnight, £9.95, Beyond.
Avalon, £7.95, Hewson.
Sherlock, £14.95, Melbourne House.

Mountains of Ket, £5.50, Incentive.
Bridgemaster, £24.95, Serin.
3D Strategy, £6.95, Quicksilver.
Super Chess III, £8.95, CP Software.
Match Point, £7.95, Sinclair.
Stonkers, £5.50, Imagine.
Mugsy, £6.95, Melbourne House.
Supercode, £9.95, CP Software.
HURG, £14.95, Melbourne House.



LOOK!

NOW THERE ARE HI-RES PROGRAMS FOR THE 16K ZX-81

3



1. Nuggets
2. Giant Rats
3. Burrowing Rat
4. Support
5. Cave In
6. Snake
7. Snake Nest
8. Gremlin
9. Mound
10. Pile of Earth
11. Cave

FORTY NINER

In 1849 the Great American Gold Rush started. Almost everyone who could sold up everything and dashed to the west coast to look for this precious metal – including you!

You must excavate this precious metal – but can you survive the giant rats and that vicious Gremlin which will come to infest your mine? Can you trick the snakes into leaving their comfortable nests and destroy the rats for you? Can you keep the Gremlin at bay?

Riches await you – but so do the hazards!

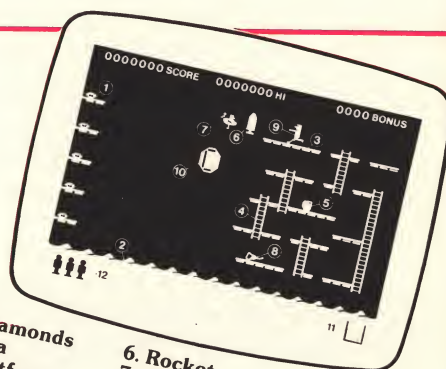
ROCKET MAN

Get rich quick by collecting Diamonds that are simply lying there waiting for you! Oh... I forgot to mention that there are one or two problems!

There is an expanse of shark infested water between you and the Diamonds and a strange breed of Bubble that seems hell bent on getting you in it! Somehow you must cross it...

You have a Rocket Pack to help you (a Vulture on higher levels) but you must rush around the platforms and ladders collecting cans of fuel (legs of lamb with the Vulture) and cursing that weird Bubble. Once you have enough fuel then it's Chocks Away!

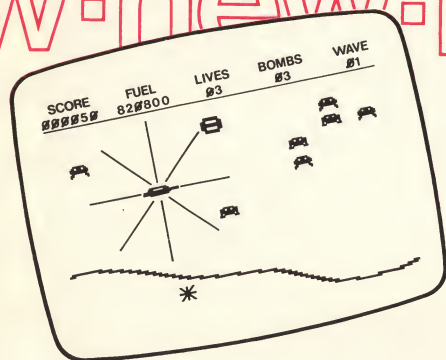
Oh... but don't run out of fuel on the way – otherwise it's... SPLASH!



1. Diamonds
2. Sea
3. Platforms
4. Ladders
5. Fuel Cans
6. Rocket
7. Vulture
8. Leg of Lamb
9. Player
10. Bubloid
11. Fuel Gauge
12. Men remaining

Z-XTRICATOR

A long time ago, in a galaxy far, far, away a terrible war took place between two hostile races. Any prisoners taken could not expect to live very long in the hands of their captors. Their only hope lay with a group of valiant warriors – the XTRICATORS – whose task it was to rescue fellow beings from the alien planet's surface. You are about to take on the role of such a warrior...



Please send me:

	QTY	TOTAL AMOUNT
FORTY NINER £5.95		
ROCKET MAN £5.95		
Z-XTRICATOR £5.95		
TOTAL		

Available from all good computer shops or send cheque/P.O. for £5.95 (inc. P&P) to:
Software Farm,
FREEPOST (No stamp required) (BS3658),
BS8 2YY.

Software Farm, 155 Whiteladies Road, Clifton, Bristol BS8 2RF.
Telephone (0272) 731411. Telex 444742 AFMADV G

Software Scene

ZX-81 software

THE ZX-81 has a great deal to answer for. For those of us who owe our living to the micro industry, be it as programmers, journalists, or company directors, the ZX-81 was the key to it all. In essence, it was and perhaps remains a far more revolutionary machine than the Spectrum or the QL.

Of course, the real kudos must still be reserved for those who actually own one of the 1K machines – if they built it themselves from a kit they certainly hit the jackpot in brownie points. But the machine is by no means a relic of the past, and over the last year the reduction in price to £35.00 including RAM pack, wobble and all, has maintained the machine's popularity in spite of the phenomenal rise of the Spectrum.

Unfortunately, the machine has not been so well served recently in terms of software. Software companies in the Sinclair market have concentrated on the Spectrum to the detriment of the earlier computer.

All is not gloom, however. A few companies whose major interest is in the Spectrum continue to release new products for the ZX-81, while a number of smaller businesses are moving in to vacate at least part of the gap left by the larger outfits, no doubt finding the competition at that end of the market somewhat less ferocious than the at times internecine strife in the Spectrum arena.

Forty Niner

Software Farm has demonstrated its commitment to the ZX-81 with a series of high-resolution arcade games, the best of which is **Forty Niner**. The game involves digging for gold, a strenuous task at best, but considerably complicated by the voracious giant rats which burrow towards you with supper on their minds.

The game's graphics are impressive, demonstrating that even without colour the ZX-81 does not have to remain restricted to character square grids with all motion held to the level of the M4 tailback on Bank Holiday Mondays.

Indeed, you may be surprised at the level of activity underground. A regular ecological cycle is involved, as the rats which prey on you are in their turn eaten by snakes, which, if you tunnel in the right direction, can be freed to devour the furry beasts.

If that was not enough, a one-legged

creature of dubious ancestry is bouncing up and down on your pile of waste earth at the top of the screen. If he gets through it is curtains, so you must return to the surface frequently to build up the mound.

Excellent fun, **Forty Niner** is addictive and well-presented, and can safely be waved at all those snooty characters who look down on the ZX-81 from the elevated heights of the dead-flesh Spectrum keyboard.

Galactic Trooper

Trigger-happy clones of Luke Skywalker will find plenty to amuse them from Romik Software. **Galactic Trooper** involves destroying one of those outrageously aggressive fleets of alien spacecraft before a single ship pollutes the world with its horrible alien culture and instant mashed potatoes.

The graphics are rather confusing, and as it is important to recognise the different types of alien you should make sure you read the instructions carefully before you play.

Galaxy Jailbreak

If **Galactic Trooper** sounds too like **Space Invaders** for your refined taste, try **Galaxy Jailbreak**. Also from Romik, the game involves guarding some alien generals, presumably captured at some stage during the previous game when you were not looking. Their loyal men (men?) are attempting to free them by eating away the walls of the prison. You of course get to beat up on the enemy troops with your stellar base. Be careful, however; if you carve up one of the generals he turns instantly into five soldiers who continue the fight, a peculiar biological feat which makes you wonder what else they might do in a tight corner. Not a lot is the answer, but in spite of the simple concept the action is fast and the control keys offer a wide choice to fit the itchiest of button-bashing fingers.

Bears in the Wood

Gentler souls will almost certainly prefer **Bears in the Wood** by Unicorn Micro Systems. You are one of the last surviving great black grizzly bears, and you have to guide your three cubs safely through the forest to your cave.

The cubs are playful little creatures which wander off the track if you do not keep an eye on them. Once separated from their mum, they start bleating

pathetically, a piteous sound which fortunately we cannot hear on the completely soundless ZX-81. The noise attracts the wicked hunters who like nothing better than bagging baby bears, and you must continually round up your cubs and attempt to get in between them and the hunters, whom you, being large and in no way pathetic, can gobble up with ease.

Once you have the cubs in the cave you can stock up your honey pots with the wild honey you have collected in the forest. There is a time limit on the operation, and if you fill all the pots in time you gain extra cubs. The game does not go into detail on this point.

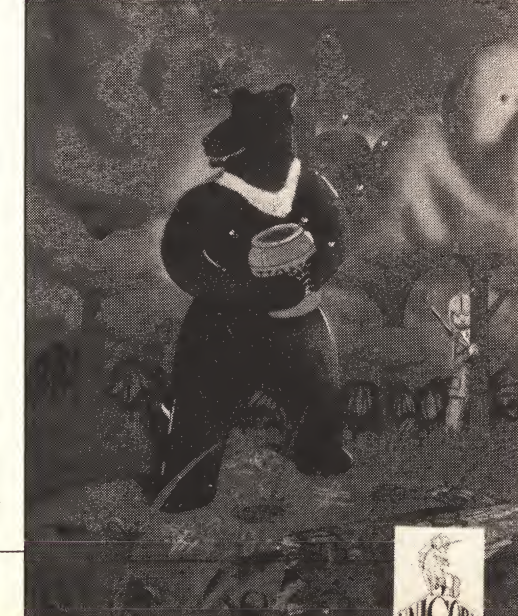
It is a pity that an otherwise entertaining and unusual game is spoiled by the poor graphics. The forest trees are represented by a capital T, the hunters by an inverse H, and the bears more like a peculiarly noisome centipede than members of "the most beautiful of the bear species" as they are described on the cassette insert. Nevertheless, the game is not without its merits, and does require a certain amount of tactical thought to play successfully.

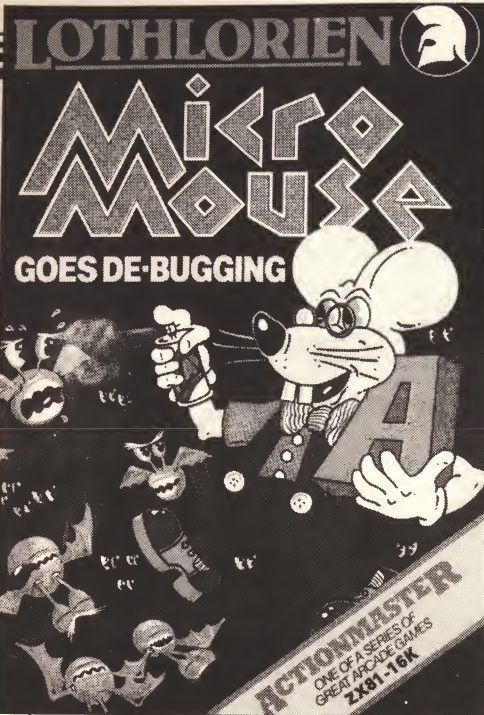
Micro-Mouse

Talking of furry animals, at the other end of the scale we have **Micro-Mouse**. Lothlorien, a company better known for its excellent range of strategy wargames for the Spectrum, has released **Micro-Mouse Goes De-bugging** for both the Spectrum and the ZX-81, a thoroughly praiseworthy decision.

The game is highly original in its form, and features high-resolution graphics. Your task as **Micro-Mouse** is to de-bug a program, which is displayed

BEARS IN THE WOOD





on the screen. Some of the letters in the statements are flashing, which indicates that the letter in question has been removed by the bugs and put in a bin at one of the four corners of the screen.

You must run round the program and in between the lines collecting the letters and putting them back in the program. Lose too many letters at the same time and the program will crash. To help you you have a can of Datakill, which eliminates 90 per cent of all known bugs for a couple of seconds, which is useful for slowing the critters down if not much else.

Volcanic Dungeon

Another way of negotiating the graphics problem is to ignore it altogether and write strategy games which require little or no graphics. Many classic adventure games began their life on the ZX-81, including the Carnell **Volcanic Dungeon** and the Artic series of adventures. Volcanic deserves recognition as the first graphics adventure for the ZX-81, and therefore one of the very first of all adventures. Unfortunately, the game mechanics are rather crude and it is frequently a matter of good fortune as to whether you survive to defeat the evil witch Magra or wind up in small bleeding pieces on the cold floor of a dank cavern.

Black Crystal

Far more attractive is the second in the series, **Black Crystal**, a massive six part adventure which represents great value for less than a tenner. You must collect the seven rings of creation and place them around the dark crystal, source of ultimate evil. The first part of the game consists of travelling round a

map of the lands in which the other programs are set, discovering the entrances to the hidden caverns and fastnesses within which the rings are to be found. Although its idiosyncratic style has won it some criticism from purists, it is one of the finest adventures available for the ZX-81, and well worth buying. Both the above programs are now available from Mastervision, a company set up by the directors of Carnell and Mastertronic when Carnell Software collapsed in mid-1984.

Pimania

Another adventure which made the successful trip from monochrome ZX-81 to glorious Spectrum luricolor is **Pimania**, the thoroughly daffy game from Automato with a £6,000 prize waiting for the lucky winner. It is now two years since the game was first launched and nobody has yet come forward to claim the golden sundial of Pi from programmers Croucher and Penfold.

Those who know the game will mutter something to the effect that nobody is ever likely to win the prize, but the game is nevertheless enormous fun or a thorough drag depending on your sense of humour and general level of decadence. TV-dinners, cans of worms, rubber ducks and the notorious Piman himself feature in what must still be the wackiest game for any Sinclair machine. We hesitate to recommend it, knowing the strong passions it provokes; you either adore Automato or you despise it. But if your neighbourhood computer store can dig out an old ZX-81 and let you try it for yourself then take the opportunity. If you have the sort of sense of humour best exemplified by Daley Thompson's T-shirts then perhaps it is about time you bought the game and found out what all the fuss was about.

Fort Apache

Strategy games are represented by a few new releases, notably **Fort Apache** from Contrast Software. The game is all about cowboys and indians, and your job is to lay siege to a fort. You have 300 men at your command, and must assign them to various tasks, such as building siege equipment or gathering supplies. You also decide the precise form of your attack — straight through the gates with the battering ram or a subtle tunnelling operation. Those with a head for heights may care to scale the walls. Whatever you choose, the computer will tell you the result, and you may

then retire to plot your next move.

There are some colourful touches to the game. If you leave too many men at your base the defenders may sally forth and attack you in the field; you must also ensure your men are fed properly, or they will begin to hanker for momma's good 'ole hominiy gits and corn pone, and desert. There are no graphics involved, but the game is an absorbing one, and it is possible to prolong the siege for some time with careful management.

It is a frequent complaint of ZX-81 owners that it is difficult to obtain software for the machine. That is particularly true of newcomers, who may be put off buying their first machine if they can see only a limited range of games available on the shelves, and cannot afford a Spectrum. It is worth remembering that just because many titles are no longer stocked by the big high street stores it does not mean the games no longer exist. Most companies still have stocks of their early ZX-81 programs, and if so, a letter of enquiry will usually get a friendly reply and price list. So there is no reason to give up hope, and if you have difficulty finding addresses and the like, you will find a list of most major software houses and their addresses in *Sinclair User* magazine, together with an extensive directory of software available for all Sinclair machines.

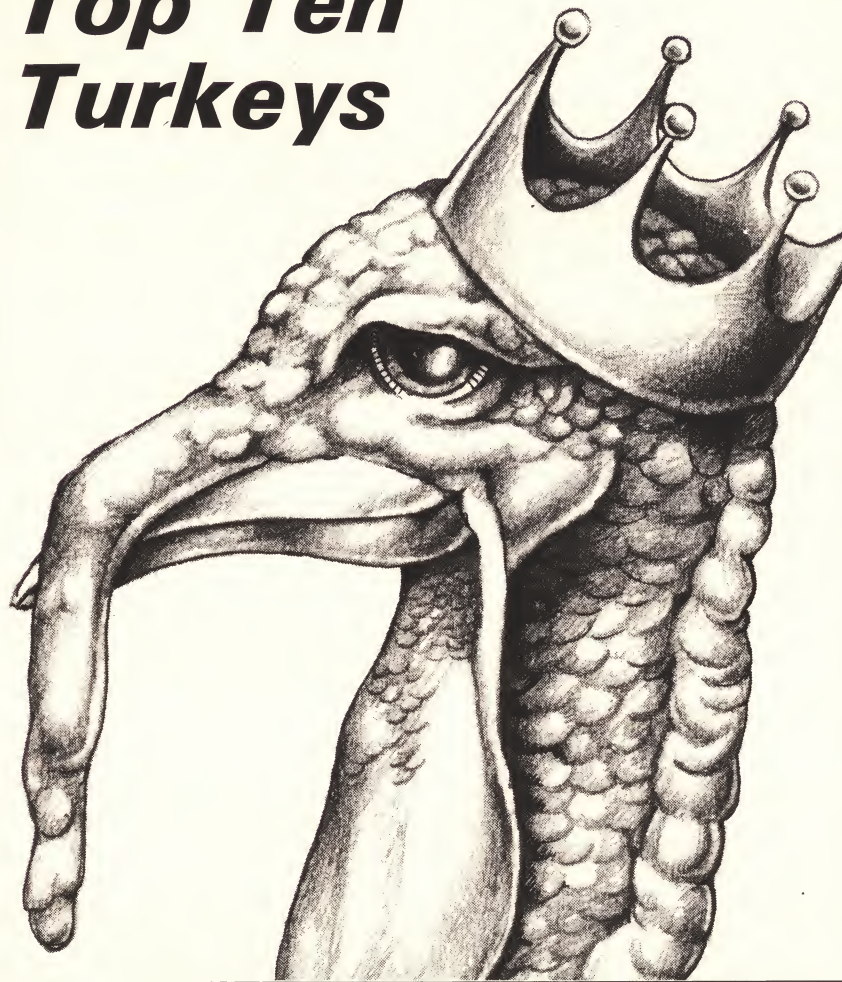
Relic of the past?

Nobody except the real hard core ZX-81 hackers can deny that the machine has passed its prime, and will eventually join the ZX-80 as a period piece. In the meantime, there are still plenty in the shops and many hundreds of thousands still in use up and down the country.

There is also good software about, and new software of quality appearing every month. No other computer has yet excelled it as a dirt-cheap introduction to computing, and one with a sensible Basic and clean, simple memory organisation. Many of our top programmers learned their trade on it, but if such application is not for you, then do not despair.

When the curtain finally falls, and you have no further use for it, you could always put it in a glass case as evidence for your great-grandchildren that you too were part of it all, way back at the dawn of history, when cars used petrol, the earth was flat, and a Sinclair month was twice as long as anybody else's.

Top Ten Turkeys



HOW DO YOU choose the worst game of the year? The software industry is capable of matching Bernard Matthews for the production of turkeys. Great flocks of the past spring to mind — **Transylvanian Tower**, where you spent hours wandering round a maze shooting the same bat over and over again. Then there was **Schizoids**, the ultimate in intergalactic rubbish, where you nudged space debris into a black hole. It was slow, it was boring, and it was black and white.

Many Sinclair users bought those games, and some used a fat percentage of a tiny dole cheque to do so. Many bought them largely because they had pretty covers and strong advertising. We all make mistakes.

So all of the games chosen look good on the outside. There are many others which could have been chosen — **Soho Sex Quest** from Malan, which besides having a repulsive theme is not even pornographic. Or what about **Advance to Mayfair** from Amazing Games, as pathetic a rip-off of *Monopoly* as you could imagine. Then there is **Empires**,

which is a fine game, but needs four Spectrums to play at any speed.

But those are unlikely to reach you unless you scour the mail-order section of magazines and buy on spec. The following products are far more likely to tempt you. If you succumb, you can hardly claim you were not warned.

Once bitten . . .

Adventure games are supposed to be exciting. **Byte** from CCS is about as exciting as a wet December night at Bradford bus station. CCS is a company that specialises in games of strategy, games which require intellectual commitment and a little bit of logic if you are to succeed. By that token, CCS ought to know a thing or two about adventure games. If **Byte** is the best idea available, the company has a lot to learn.

You play the role of a computer. Instead of being given statistics based on strength or intelligence, you get a value for your RAM or the speed of your processor.

That is not a bad idea. It is certainly not a particularly good idea, but let your

innate prejudice against computer games about computers pass for the moment.

The real trouble with **Byte** is that it is slow, boring, and simplistic. Every time you type in a command the Spectrum plays a tedious jingle, lists all your statistics, which you should know anyway, and then tells you what you just typed in. Evidently the game is aimed at morons, not *Sinclair User* readers. Finally, you may be treated to a staggeringly imaginative description of your location, such as, "You are in room 7,4. You can see: Box!"

The layout of the game is a simple grid structure with several levels. The only surprise worth mentioning is an instant-death trap which pops up at random and sends you back to the beginning with all your circuits scrambled. It is an effective device to persuade you not to try again.

Byte is an important landmark in the art of changing good money for dreadful software. It may not be the worst adventure game released, but it might be the one which will cause you the most sorrow.

Draw the line

The idea behind **Close-In** from Pulsonic is not original, but it is one of the few new ideas in games over the past few centuries. That makes it all the more criminal that Pulsonic should have made such a great effort to produce the most boring version imaginable.

Close-In requires that you draw a line around the screen while the computer or a human opponent does likewise. If you cross your own or your opponent's line you lose. Although very simple, there is great potential for strategic thought based on spatial concepts, and with attractive presentation the game could be a success.

Unfortunately, the line remains a line, one pixel in width, on one of five different screens, four of them simple mazes, and the other one completely blank. There is not even a background colour to liven things up.

Furthermore, the computer cheats, which is definitely not what one expects from one of Sir Clive's machines — dash it all, the man's been knighted! It has a nasty habit of breaking off in mid-game and starting a new line somewhere else. Needless to say, that privilege is reserved for the computer.

To round off the game Pulsonic has thoughtfully chosen not to allow the use of joysticks, which means you have to

fiddle about with eight directional keys to move the line. With two players, that makes sixteen keys on a cramped keyboard, hardly likely to prove easy.

Admittedly the game only costs £2.95, but it is still a rip-off. The disappointment is exactly the same emotion whether you pay three quid for the game or thirteen. It is only the violence of your outrage which changes.

Battle weary

Many people enjoy wargames. They enjoy pushing soldiers around a tabletop, measuring angles of fire and throwing dice to determine casualties. A fair number of people enjoy pushing cardboard counters around sheets of paper marked out in hexagons to complicated rules. The one complaint most wargamers have is that the games usually drag on for hours, most of that time being taken up searching for the appropriate rule in a 200 page book.

Computers ought to take the pain out of wargaming, and leave you to enjoy the vicarious thrill of massacring large numbers of people. Not so in **Reichswald**, from MW Gamesworld. A simulation of the Battle for the Rhine in 1945, you shunt your pieces around a grid-map and wait for the computer to make its move and resolve combat. The computer is very slow, and results are usually the deduction of a few points off the combat strength of your units. That turns the game into a war of attrition, no doubt very realistic, but not the stuff of an exciting computer game.

If the structure of the game requires such a slow rate of play then it is inappropriate for the computer, and MW Gamesworld should have looked for something which would run faster. That does not necessarily imply a simplification; there are many games of enormous complexity which run at a very high speed.

MW Gamesworld rather pompously states on the cassette cover that it is not its purpose to glorify war. The authors should not have worried. There is nothing glorious about Reichswald.

Tame tanks

In spite of its quite revolting title, reminiscent of the very finest video nasties, **Driller Tanks** is a tame little number from Sinclair. It can be given to the most impressionable toddler with no qualms at all. It might even help get him to bed on time.

Your task is to burrow underground in order to deal with various monsters seeking to invade the surface world.

The monsters look like giant crabs. What giant crabs are doing underground is unknown, but the answer is probably as logical as the weapon you have to fight them off — a tank with a Black and Decker on the front end.

Graphics are large and somewhat crude, and the joystick controls, while simple, do not respond with great sensitivity. The screen is badly organised in that much of the top half is taken up with a garish building looking vaguely like the Taj Mahal which you are supposed to be defending. You might sooner defend an extension to the National Gallery.

It is ironic that Sinclair is capable of creating extraordinary hardware products and yet seems unable to distinguish between good and bad software which, as in the present case, it usually buys in from other companies. Amongst undoubted gems there is far too much dross in the Sinclair catalogue. **Driller Tanks** is appalling, and at £5.95 outrageously overpriced.

Dodgy program

Creative Sparks has provided a tremendous incentive to would-be programmers with the game **River Rescue**. It is one of those games which inspires you to great deeds — you just know you could write better yourself.

Briefly — and to be honest there is not much to say about the game anyway — you have to pilot a fast gunboat along a river. You must avoid logs and crocodiles, either by dodging them, or by blasting them out of the water.

Instructions are kept to a bare minimum. No advice on which keys to press to start the game, nor how to obtain joystick options. That is an advantage. You will probably have more fun working out what to do than you will once the game starts.

Dodge-it games, where the screen scrolls while you move from side to side to avoid obstacles, are frequently the sort of games which form one's first attempt at writing an arcade game. Unless you are planning to augment the game with very classy graphics and a wide variety of levels with their own special problems it would be better for your commercial health to avoid them like the plague. Of course, most amateur games of the type scroll up the screen. **River Rescue** scrolls sideways, which indicates someone at Creative Sparks has read a book on machine code, but hardly makes the game any more interesting to play.

It is one of the most unoriginal games

of the year, with clumsy graphics and little detail. You will probably heartily agree with the game's subtitle, 'Racing against Time'. Loosely translated, it could be taken to mean, 'How many copies can we sell before the awful truth is known?'

Luke Sleepwalker

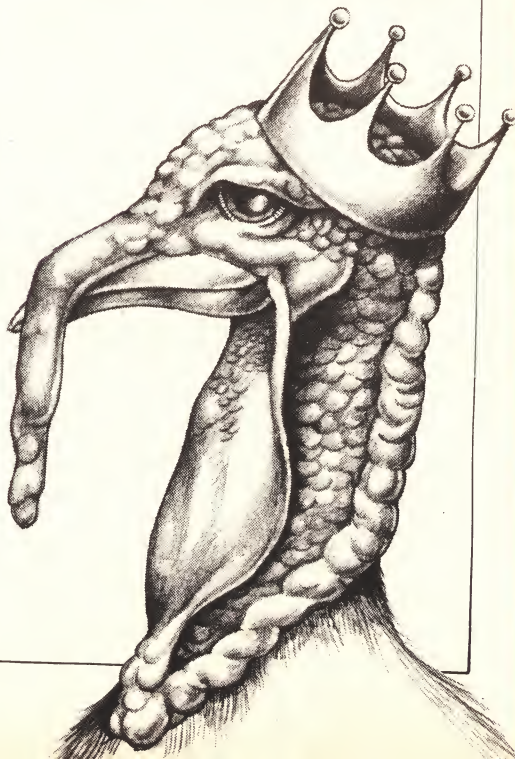
The turkey has landed with full force in the guise of **3D Star Wars**, from Add-on Electronics, which bears as little a resemblance to the Lucas film as you could possibly imagine. The publishers obviously realised that too, and on the cassette inlay changed the name of the hero to Luke Clearthinker, in the hope that the customer would not be misled. Unfortunately they forgot to inform the programmer, who throughout the game instructions refers to Luke Skywalker.

The term 3D is a bit of a misnomer too, as is apparent when the mess of coloured blobs which represent the enemy craft start to dart at you from all directions. A confusion of keys makes firing more a matter of luck than judgement. There are, apparently, several levels of play; you will not be encouraged to explore them.

Leaden game

Pac-man rears his ugly head again in **Magic Meanies**, a tedious piece of arcade drivel from CDS Micro Systems. Your moveable feaster, Meltec, runs around collecting lead and cherries, a curious diet.

He is pursued by the deadly boring meanies, which follow him along the maze-like path he has created in his travels. Fortunately our hero has balls,



crystal ones in fact, which he can fire at the encroaching meanies. Achieving a high score is no problem, but the author has left the most difficult part until last, and entering your name on the high score table is likely to leave you dazed and confused — a hollow victory.

If you cannot find a copy of **Magic Meanies** in the shops, don't despair; most computer magazines publish listings of programs which can match, and even surpass, **Magic Meanies** in originality and speed.

No dice

What complex task is a computer best employed in? Why, simulating the throwing of dice of course! Bug-Byte fully explores that enormous potential of the 48K Spectrum in its simulation of the electoral process, **General Election**, a game which condenses all the monotony of the real thing into a concentrated yawn for two to four players.

You set off on the campaign trail attempting to woo the local populace, visiting 100 constituencies and moving according to the dice cast by the computer. Gripping player involvement is included, too, when you have to decide what the best policy might be — similar to letting the computer think of a num-

ber which you then have to guess.

The game concludes with a blockbuster of a finale when the election results are displayed. That takes almost half an hour; for added realism you could play the game at three in the morning and fall asleep while the results are read out.

Darts bored

Another simulation — and as with most computer simulations the term is used loosely — is **On the Oché** from Artic Computing. This 'thrilling' game brings you 'all the excitement of competition darts'. Of course, it doesn't do any such thing, and any darts player worth his salt wouldn't look twice at it.

Aiming the feathers at the board is simplicity itself. It would be possible to score 180 every throw if it wasn't for a seemingly random crosswind which blows across the pub every now and then, and for which you can do little to compensate. Skill is consequently replaced entirely by luck. In that way the game accurately simulates the act of closing your eyes and sticking a pin into a piece of paper. Thrilling.

Terriblehawks

The last great software disaster of



1984 is **Terrahawks**, published by CRL. As we have seen with **3D Star Wars**, the golden rule to follow with film or TV software spinoffs is to ensure the game bears no resemblance whatsoever to the source material. CRL have come up trumps with **Terrahawks**, which mentions neither the characters nor the setting of the popular children's TV series, though some passing similarity to the title sequence might be noted.

Title aside, the game is sub-standard shooting and bombing arcade fare as you guide your craft through the maze of high-rise flats economically outlined against the void. Buildings can be avoided or bombed, or eliminated completely by switching your Spectrum off and playing something entertaining instead.

Unbeatable value.

Almost unbeatable games.

CRIBBAGE
Available on the Spectrum for the first time. **CRIBBAGE** is an excellent version of the popular card game. As well as superb graphics, the game includes full rules and playing instructions together with automatic scoring and score re-view.

MADHATTER
"I must get this place tidied before my guests arrive," wails the **MADHATTER**. Help him collect his tea time treats for his Un-Birthday Party. "But watch out for those mice!" he warns, "and keep away from those vacuum cleaners!"

ORC SLAYER
In your search for Thor, leader of the Death Moon Tribe of Orcs you must first find the Magic Runesword needed to destroy him and then seek out their underground stronghold somewhere in the Savage Lands.

Full of excitement, this adventure will push you to the limit of your wits and intelligence in achieving your goal.

GAMMA SOFTWARE

Please send me the following games for the Spectrum:

MADHATTER 48K @ £5.50

ORC SLAYER 48K @ £5.50

CRIBBAGE 48K @ £5.50

Total including postage: £

Name _____ Address _____

GAMMA SOFTWARE
12 Milverton Rd,
NW6 7AS

Available from Leisure Soft, Center Soft and all good retailers

Seeing is believing

See the pack – see the screen – see the game.
See the difference the moment you pick up one
of the games from Firebird's new Silver Range.

The difference is that our packs show the high
quality you'll see on the screen when you play,
so you know exactly what you're buying.

Here's a sneak look at three of the NEW games:



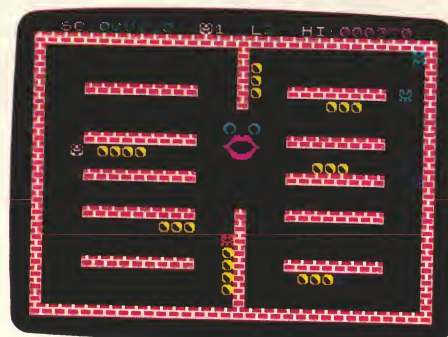
VIKING RAIDERS

THRILL to the Vikings' exploits in this highly original war game. SAIL the Fjord in search of treasure and adventure. LEAD your armies in raids on enemy castles. CURSE at the uncontrollable antics of drunken Norsemen. MAY ODIN PROTECT YOU! For 1, 2, 3 or 4 players (any combination of human and/or computer controlled Viking armies).



EXODUS

Galleons, abominable snowmen, hover mowers, televisions and llamas are just some of the strange characters trying to escape from The Pit. Your task is to zap them with a laser bolt before they reach the rim. Whatever you do, don't shoot Spud!



CRAZY CAVERNS

Ten Crazy Caverns, ten hungry mouths to feed. Line up the Crunchy Cubes and cannon them into the Craving Craw. Watch out for the Crazy Crashers coming to convert your little Cubey Crammer into a Crushed Crisp. It's a cracker!

The same high standard applies to all twenty
games in the range and all are available at the
remarkably low price of

£2.50



Firebird

SOFTWARE



The new range from British Telecom.

You'd better believe it!!!

FIREBIRD SOFTWARE, WELLINGTON HOUSE, UPPER ST. MARTIN'S LANE, LONDON WC2H 9DL. TEL: 01-379 6755/5626
Firebird and the Firebird logo are trademarks of British Telecommunications plc.

Look out for these games in the FIREBIRD silver range at selected high street stores.

SPECTRUM: THE WILD BUNCH · BOOTY · MR. FREEZE · TERRA FORCE · VIKING RAIDERS · MENACE · EXODUS · RUN BABY RUN · CRAZY CAVERNS.
COMMODORE 64: BOOTY · MR. FREEZE · EXODUS · HEADACHE · ZULU.

VIC 20: SNAKE BITE · MICKEY THE BRICKY.

BBC MICRO B: BIRD STRIKE · GOLD DIGGER · ACID DROPS · DUCK!



Available for B.B.C. Model "B",
Electron, Spectrum and Commodore 64.
Espionage is available from all leading
High Street Stores and quality Computer
Games Specialists.

price **£8.95**

Retailers contact Mr. Len Fisher on
our telephone hot-line (0642) 227223.

Espionage

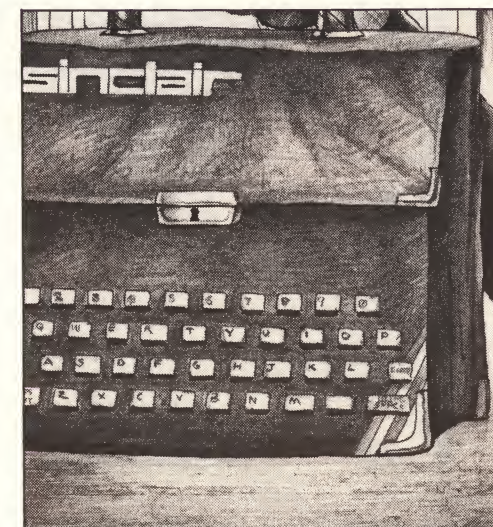
MODULAR RESOURCES LIMITED



A GAME OF INTRIGUE & SKILL IN THE OIL BUSINESS.

Chapter 6

Business



With the exception of the QL,
the range of Sinclair computers
is not ideally suited for use in business.

Both the ZX-81 and the Spectrum
are, however, extremely cheap
and thus within the reach of the small businessman
with limited resources.

Mike Wright provides an expert introduction
to setting up a business system,
and explores the pitfalls which lie in wait
for the unwary and the hasty.

THE COMPUTER revolution is well and truly with us. The benefits of the microchip are available to anyone for a very modest outlay. But, having played **Jet Set Willy** or **Chuckie Egg** until your wrists and fingers ache, perhaps the time has come to do something with your Spectrum or ZX-81.

Far from being simply marvellous games machines the Spectrum and 16K ZX-81 are also capable of helping at work. The uses they can be put to extend beyond easing the boredom of paper work, and they are capable of running advertising displays, switching lights on and off, deterring burglars and even controlling manufacturing equipment.

However, introducing a computer into your business is not going to cure all ills without some effort and expense on your part. Before setting off on the road to computerization very careful thought needs to be given, firstly, as to whether it will improve your business and secondly to determine which areas of your work will benefit from its introduction and which areas will be better left in their present state.

Why and when

Often the decision to use a computer is made because the person making the decision either 'knows' that computers are more efficient or because he has one at home, enjoys using it and wants to extend his enjoyment to work. In such cases not enough thought may be given as to how efficiency will be improved and how the company will benefit. Efficiency is not going to be improved if, for example, you as a manager use the computer to do tasks that would otherwise be done by junior staff.

Not all areas of work are suitable for computers and even areas that are suitable in one company may not be in another. If your cash books shows only three or four entries per day then it is likely to be more efficient to continue using a manual system. Computers are most useful when applied to doing repetitive tasks such as processing payrolls or handling large amounts of data. Once the basic details of a payroll have been entered it can often be produced in a fraction of the time.

Even if you already have a home computer probably you will need to buy equipment if it is to be used in your business as well. Computer equipment falls into two distinct categories, hardware and software.

Hardware refers to the actual ma-

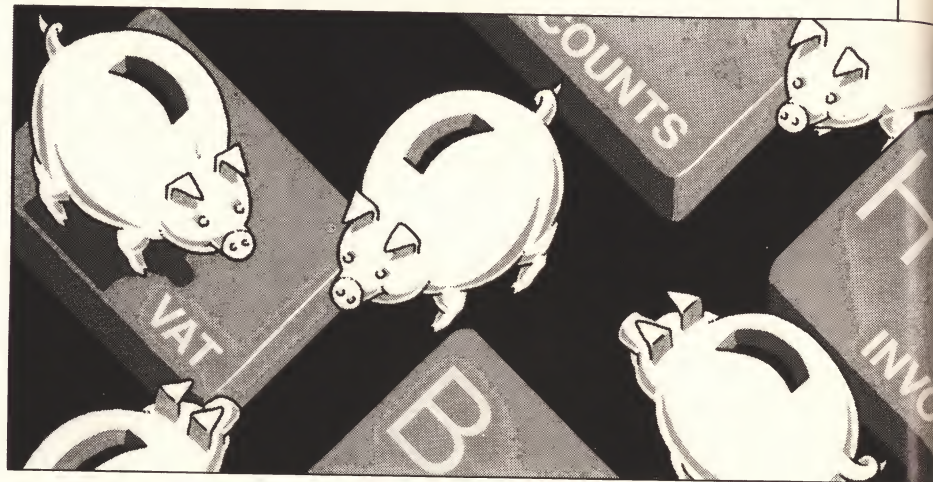
chinery you use. For use at work you will need in addition to the computer, a means of displaying the results, a method of storing your data, a printer to produce a permanent (hard) copy of relevant items and a real full-travel keyboard.

Display

For the display you will need either a 625 line television or a monitor. Both the Spectrum and ZX-81 are designed to be used with a television. Although televisions are satisfactory for most uses some programs now display more than

recorded onto cassette tape unless the cheapest and poorest quality tapes are used. The major disadvantages lie in the fact that only serial access is possible. That means programs and data can be stored and read only in sequence on the tape. Moreover, the rate of data transfer from the tape to the computer memory is very slow. Both factors make the loading and saving of large blocks of data or programs very tedious.

The most popular storage device for business use is probably the floppy disc. The disc is similar to a small record but instead of the head having to follow the



the standard 32 characters per line. The result, depending on the make and age of the television, is often a display that is very difficult to read and which can lead to eye-strain. Some televisions have been found to be unsuitable for certain computers and you will need to seek your dealer's advice or at least arrange to return the set if it does not work.

Monitors are more expensive than televisions but give sharper, clearer pictures which are easier to read and most have anti-glare screens. The extra amount spent on a monitor is certainly a good investment for the business user. Some monitors such as the Microvitec Cub are supplied with an adaptor to convert the signal for use. If your monitor does not have an adaptor it is possible to bypass the signal modulator in the computer. That however invalidates any guarantee and should only be attempted by those who are fit to hold a soldering iron.

Storage

The most common method of storing data and programs on home computers is cassette tape. The advantages lie in the cost — a cassette recorder costs as little as £16.00 while the tapes cost less than a pound each — and the reliability. It is unusual to lose data which has been

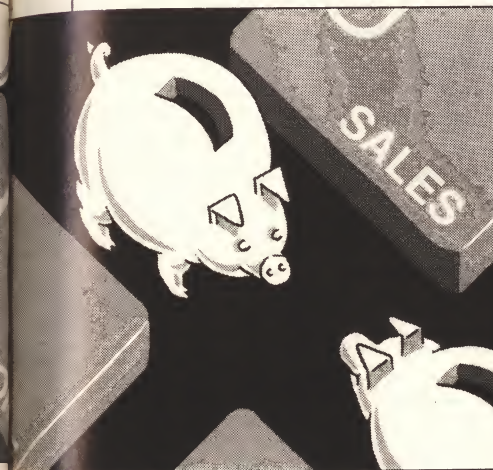
grooves it can move backwards and forwards across the disc which is spun at high speed. That allows it to read bits of programs from different parts of the tape as if they followed one after the other. That is known as random access. Although reliable and fast discs are at least to begin with, an expensive method of storage. The cost of a single disc drive starts at about £200 and each disc will cost a further £1.50 to £4 each. Apart from the cost another disadvantage is the lack of software which is ready to run, or can readily be adapted to run, on discs.

For Spectrum users other options exist with the ZX microdrives and the

Rotronics Wafadrive. Those devices use cartridges called stringy floppies. They consist of a continuous loop of video tape spun at high speed.

To use a microdrive which costs £49.95 you will also need the ZX Interface 1 which is a further £49.95. Although microdrives offer reasonably fast mass storage they have only serial access and there are still doubts about their reliability. A particular problem lies in trying to load data save on one microdrive from another.

The Wafadrive has been launched only recently, costs £129.95, and has



with a very powerful word processor included in the cost. It is reported to be almost as reliable as a disc and to allow a pseudo random access by spinning the tape at different speeds. However, it still takes slightly longer to load than the microdrive.

When you settle down to using your computer you will find there is a lot of loading, saving and verifying to do. In some cases, particularly when using cassette tape, more time is spent doing this than would be spent on doing the job manually.

Hard copy

As you will need some kind of perma-

nent record, especially for accounting trails, you will also require a printer. There are, in effect, three different types — the daisy wheel, dot matrix and thermal. A daisy wheel prints in the same way as a typewriter. It produces the highest quality print but generally is slower and costs more than other printers. A popular buy is the Juki 6100 at around £400.

A dot matrix printer uses a system of pins in a rectangular array. Characters are formed by different arrangements of the pins striking the ribbon. As a rule, because each character is composed of a series of dots, print quality is not as good as that given by the daisy wheel but dot matrix printers are faster. If you intend using one to send printed material you will need one with true descenders. Those print letters such as p and q with their tails below the line. Prices start around £200 and a high quality model will cost about £375.

Thermal printers operate by etching the characters onto the paper. The print quality is usually only suitable for internal work and they are slower than dot matrix printers. However they are the cheapest with prices as little as £60.00. A review of printers can be found in the hardware section of this publication.

Both the Spectrum and ZX-81 cannot be linked directly to a printer as they have no built-in Centronics or RS232 port and consequently you will need to buy an interface to connect them. Centronics interfaces are made by several companies including Kempston, Tasman and Hilderbay and cost between £40.00 and £50.00.

Keyboards

A major criticism of Sinclair computers has always been the keyboard. The touch-sensitive membrane of the ZX-81 and the rubber buttons of the Spectrum both require slow and accurate pressing of keys to ensure data is entered correct-

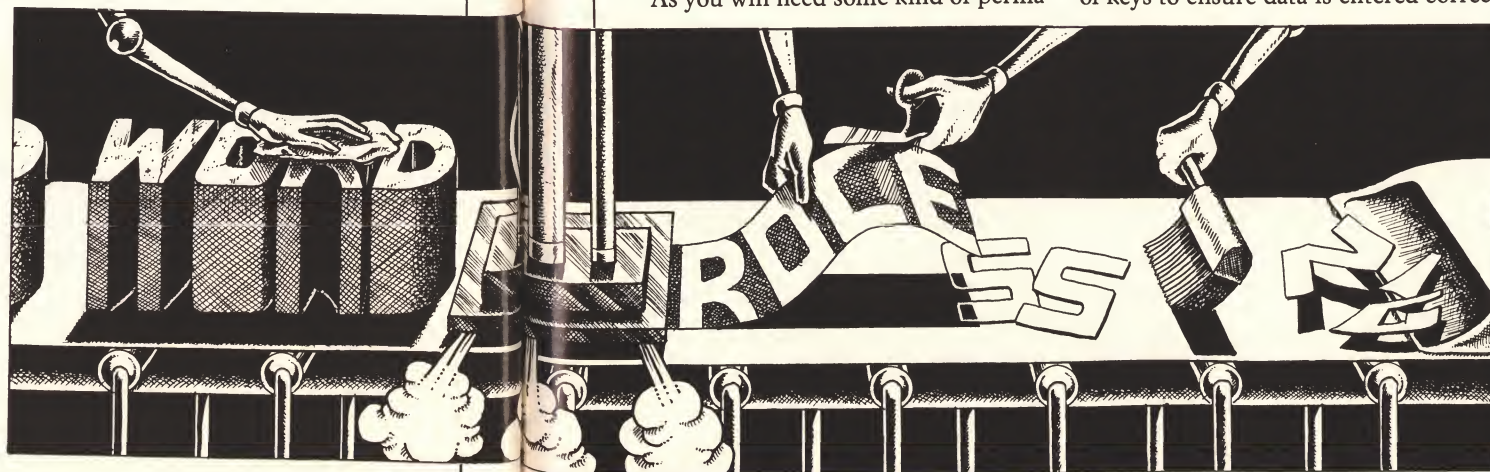
ly. A wide range of keyboards is available for the Spectrum with each offering different facilities. Your choice will depend on the facilities you require and on how much you are prepared to pay. Prices range from £45.00 to £70.00. A review of keyboards is also printed elsewhere in this issue.

If I was to use a Spectrum in business and cost was no problem then probably I would choose the following hardware. A 48K Spectrum (£130), a Microvitec Cub colour monitor (£230), a Transform keyboard (£65.00), a Sprint cassette recorder to load programs four times faster than other models (£65.00), twin microdrives and interface 1 (£140), an Epson FX80 printer (£350) and a Kempston Centronics type printer interface (£45.00). The total cost for the system is over £1,000. It can, however, be done for half the price, by using 48K Spectrum (£130), a Pye black-and-white television (£50.00), a Transform keyboard (£65.00), a Walters WM2000 printer (£200) and a Kempston interface (£45). The total cost is about £500.

Software

Having looked at the hardware you should also look at the programs (i.e. software) which are available. Software comes in two forms. There is ready-to-use commercial software, and there is also your own software written for a particular task. If you decided to write your own then you should be aware of the pitfalls. The first problem you will encounter is defining the task in sufficient detail so that a program can be written. Then there is the writing and editing of the program. That may be relatively simple depending on the work done initially but it must be followed by a period during which the program is exhaustively tested for bugs and accuracy.

A program with even the most occasional of bugs is of no use if important data is lost when it is used. Once the program has been written and debugged it will probably require rewriting for use by other people. That will involve writing a manual of some sort. The program is not going to be much use if you are the only one who can use it. What are you going to do during holiday periods or sudden crises; close the business? It is not unknown for businessmen to start writing their own programs only to find they are still improving them years later or even that when completed they do not do what is required. If you are thinking of writing your own programs stop and look care-



fully for a commercial program which meet your needs or could be adapted to do so. It could save time and money.

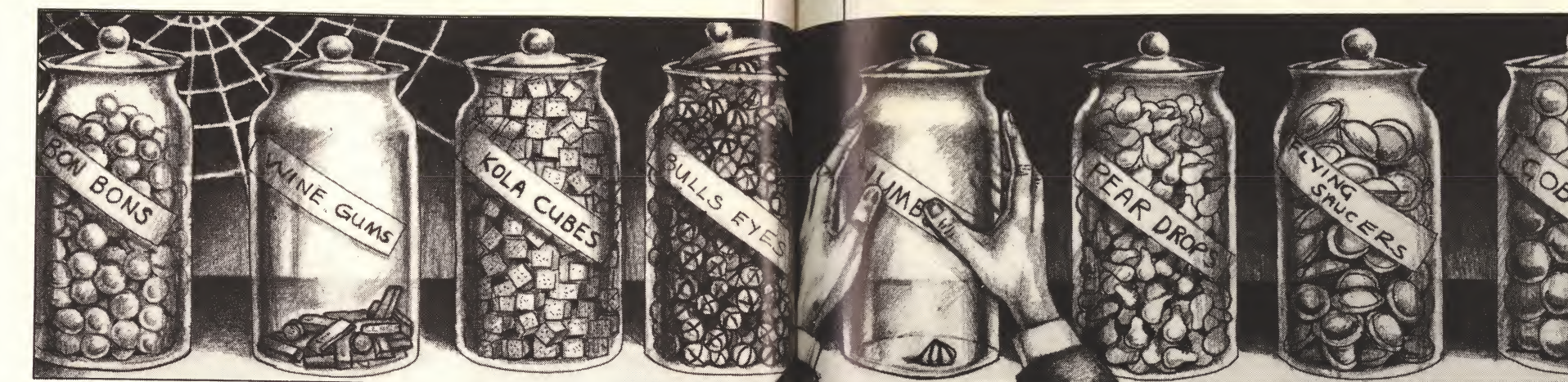
Commercial software is available for a wide range of business functions. The majority of software is for the 48K Spectrum or ZX-81. The cost of business software is also important and prices for Spectrum programs seem expensive when compared with that of games software. However, it is only a fraction of the cost of software for 'serious' computers and in many instances is as good as the more expensive programs. The quality does vary enormously and even the best program may not do exactly what you want. In general, you should read the reviews in the business section of *Sinclair User* and if possible find a local dealer who will let you try the software in the shop before you buy.

Word processing

The three most widely used types of business software are the word processor, the spreadsheet and the database. Word processing is the term used to describe the editing of any written work. A word processor should allow you to type in text, insert and delete words and phrases, move blocks of text around, print it and then save it for use later. That gives you the opportunity to alter the text so that it makes the maximum impact on the person receiving it, all before it is printed. More advanced word processors will allow letters to be personalised with the names and addresses taken from a separate file — a mailing list. While they are most commonly used for drafting letters, word processors can also be used for address labels, mailing lists, invoices, reports and any other work where text is regularly used or edited.

Of the many word processor packages available for the Spectrum **Tasword Two** with **Tasmerge** from Tasman Software is probably the most powerful available although **Spectext** from McGraw-Hill should also be given some consideration. Both have facilities for mailing lists. If you have only a ZX type printer then look also at **Micropen** from Contrast Software. **Spectral Writer** from Softek also looks promising, though at first available only with the Wafadrive; a tape version was to be released towards the end of 1984. For those who use a ZX-81 **Tasword** is available from Tasman Software.

Spreadsheet programs mimic a large sheet of paper printed with rows and columns. Those can be used for text or



data and allow you to draw tables easily and quickly. In addition the programs allow the contents of a box to be calculated from the contents of other boxes by the use of formulae. One figure can therefore be changed and the rest of the spreadsheet recalculated quickly if not automatically.

That makes them ideal for use in forecasting. Spreadsheets are mostly used for financial applications although they can be used wherever figurework is involved. Spreadsheets to consider are **Omnicalc** from Microsphere and **VuCalc**, which is also available for the ZX-81, distributed by Sinclair Research and produced by Psion.

Databases

Databases are used to store and retrieve information. They vary in complexity from those which duplicate a simple card index to more powerful programs which require a certain amount of programming ability to be used to their best advantage. Databases are made up of fields. Those are the individual pieces of information in the simplest form and can be thought of as a single line on a card. A group of fields is called a record and can be imagined as the whole card. All the records form the database which would be the card-box. If you wished to create a database of debtors then individual fields would consist of names, addresses and amounts owed. The data on an individual would be a record. The simple card index type of database enables all that information to be kept but the more powerful ones will also allow, for example, the amount owed to be totalled for the entire database.

Of the many databases produced for the Spectrum **Masterfile** from Campbell Systems is the most powerful and the one I would recommend even for use simply as a card index. Of the card index type **Vu-File** from Psion is well

worth looking at. A version of that is available for the ZX-81, as is the predecessor of **Masterfile**, **The Fast One**.

Applications programs are released regularly with the majority of requiring a 48K Spectrum at least, and most of those are for financial applications. There are a number of ledger programs, either general or specific, such as sales and purchase ledgers. If you are looking for a suite of ledgers then those can be obtained from Transform, Hestacrest and Kemp. There is very little to choose between any of them. One general-ledger which stands out above the rest is **Finance Manager**, or **VAT Manager** if you need VAT details, from OCP.

Miscellaneous

I have mentioned already the payroll as one area that is likely to benefit from computerisation and programs to cope with that are available from V & H Computing, Hilderbay and Transform. Each has its own features and it is important that you examine carefully how they operate before you finally commit yourself. The Hilderbay program probably has the most features but any pay other than regular hourly pay must be entered as one single sum, while the V & H program will handle four times as many employees as the others (200), though it does not cater for 'Contracted out' NI contributions. A full review of those programs was published in the July issue of *Sinclair User*.

Other areas in which programs exist include stock control from OCP, Kemp, Transform and Hilderbay; Statutory Sick Pay from Hilderbay; business graphics from McGraw-Hill; mailing lists from OCP; and keeping records of stocks and shares from Michael Slatford Software and Morley Davies Associates. Programs are also available to run an advertising display, **DLAN** from Campbell Systems; calculating beam strengths, from Beamscan;

costing building materials, from J Redman; critical path analysis, from Hilderbay; and statistics, from University Computing and Bridge Software.

The possibilities

Sinclair computers are used in many different sectors of the business community ranging from sweetshops to sermons and from cows to cabs. A minister in Newcastle used to spend hours sorting his rota of preachers. Schedules needed to be worked out three months ahead. The problems of arranging a rota of forty preachers to cover sixteen churches, allowing for holidays, illnesses and travel difficulties and providing variety for his congregations was time consuming. His answer was to write his own program to do the scheduling on his ZX-81. The whole process now takes only a quarter of an hour.

A farmer in Wales is using a program

to manage his herd. The program runs on a 48K Spectrum and was developed by his son. Information on individual cows in a herd of 150 enables him to check on items such as feeding and milk production. A drop in output can quickly be spotted and corrective action taken. The program takes under an hour to run fully. That may seem a long time but it is 10 times faster than doing the job manually, a big bonus for any busy farmer.

In Hampshire, a fleet of private hire cars is being run by two Spectrums. One is used for calculating fares while the other looks after the administrative details such as letter writing. The company uses the Tasword word processor. A company in Stockport involved in engine tuning and carburation problems uses the Spectrum, microdrive and Seikosha printer for business and financial control. The company is another

which uses Tasword to produce letters and other written documentation. The system provides a summary of the week's trading within half an hour of closing for the weekend.

Beamscan, 20 Vaughan Avenue, London NW4 4HU.

Bridge Software, 36 Fernwood, Marple Bridge, Stockport, Cheshire SK6 5BE.

Campbell Systems, 57 Trap's Hill, Loughton, Essex IG10 1TD.

Contrast Software, Farnham Road, West Liss, Hampshire GU33 6JU.

Hestacrest Ltd, PO Box 19, Leighton Buzzard, Bedfordshire LU7 0DG.

Hilderbay Ltd, 8/10 Parkway, Regents Park, London NW1 7AA.

Kemp Ltd, 43 Muswell Hill, London N10 3PN.

McGraw-Hill Ltd, Shoppenhangers Road, Maidenhead, Berkshire SL6 2QL.

Microsphere, 72 Roseberry Road, London N10 2LA.

Morley Davies Associates, 11 Denham Lane, Chalfont St Peter, Buckinghamshire SL9 0ER.

OCP Ltd, 4 High Street, Chalfont St Peter, Buckinghamshire SL9 9QB.

Psion Ltd, 22 Dorset Square, London NW1 1UP.

J Redman, Stevelon House, Slade Lane, Tarnock, Axbridge, Somerset BS26 2SH.

Michael Slatford, 3 Campden Road, South Croydon, Surrey CR2 7EQ.

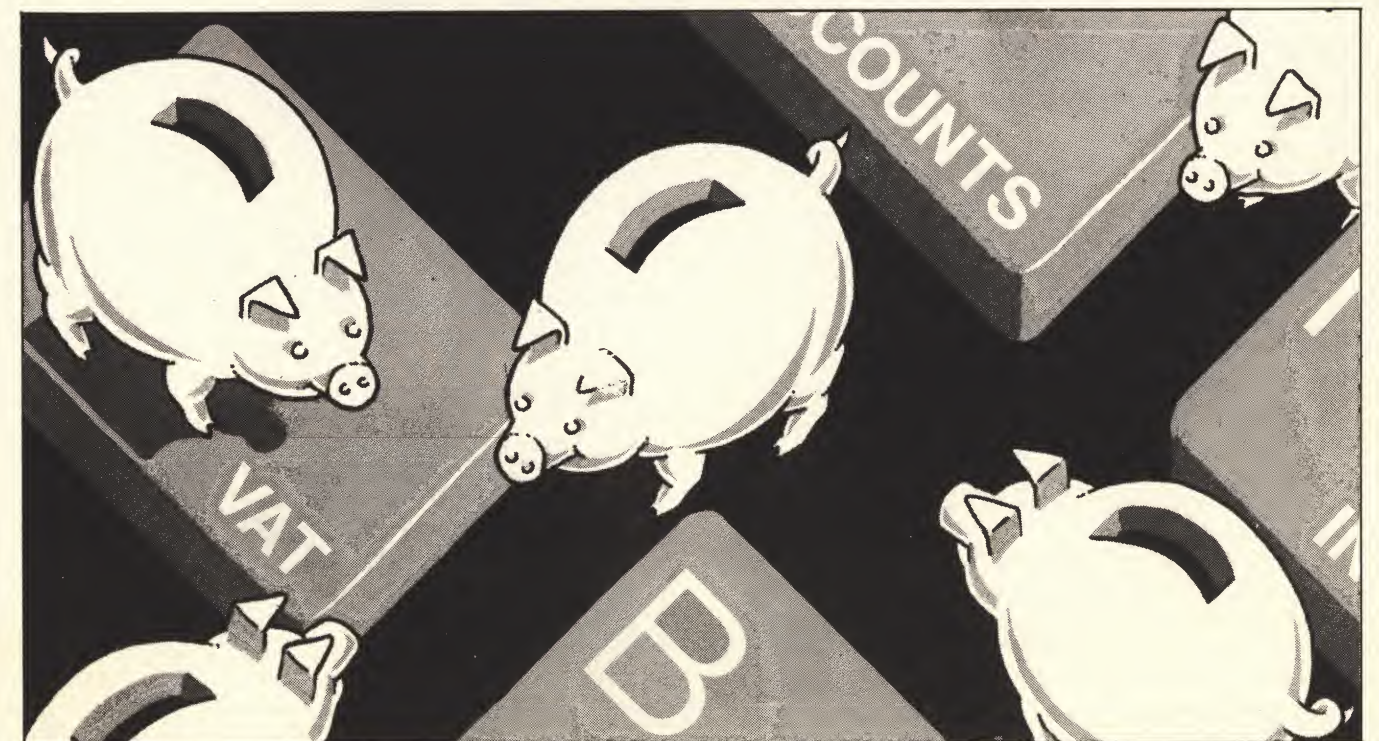
Softek Ltd, 12/13 Henrietta Street, Covent Garden, London WC2.

Tasman Software, 17 Hartley Crescent, Leeds LS26 2LL.

Transform Ltd, 41 Keats House, Porchester Mead, Beckenham Kent.

University Software, 29 St Peter's Street, London N1 8JP.

V & H Computer Services, Mayfield House, Spencer Street, Bognor Regis, West Sussex PO21 1AP.



CURRAH μ SPEECH



Speech Synthesiser for ZX Spectrum

The **CURRAH μ SPEECH** is ready to talk immediately on power-up, has an infinite vocabulary and outputs speech and ZX Spectrum sound through your TV speaker. There is no software to load with **μ SPEECH** — sophisticated Gate Array technology means you can just plug in and start constructing words and sentences like this:

LET SS = "sp[ee]k [nn][oo] [ee]vil" will say "speak no evil"! Further commands control the "voicing" of keys as they are pressed, and an intonation facility allows you to add expression to the speech.

μ SPEECH is fully compatible with ZX Interface 1 and may be used with the **CURRAH μ SLOT** Expandable Motherboard, allowing easy expansion of your ZX system. **μ SPEECH** and **μ SLOT** will also be compatible with the **CURRAH μ SOURCE** unit when it arrives later this year, allowing you to write **Assembler** and **FORTH** statements directly into your **BASIC** programs!

Top selling games like **ULTIMATE'S Lunar Jetman** feature **μ SPEECH** voice output — watch out for other titles from Bug-Byte, CDS, Ocean, Quicksilver and PSS.

μ SPEECH is available from COMET, W.H. SMITH, WOOLWORTHS, GREENS, BOOTS, JOHN MENZIES, SPECTRUM STORES and good dealers nationwide — or use the form to order the **CURRAH μ SPEECH** — winner of the CTA 'Product of the Year' award 1984.

CURRAH

To: MicroSpeech Offer, P.O. Box 1, Gateshead, Tyne & Wear, NE8 1AJ

Please SupplyMicroSpeech unit(s) at £29.95 each incl. VAT & P & P

.....MicroSlot unit(s) at £14.95 each incl. VAT & P & P

Name (please print)

Address (please print)

Postcode

I enclose a cheque/PO payable to 'MicroSpeech Offer' value £

or debit my Access/BarclayCard No.

Cardholder Signature

Credit Card Hotline 091 - 482 4683 Please allow 28 days for delivery. Offer valid UK only

Chapter 7

Machine Code



Sooner or later most Sinclair users become frustrated at the confines of Basic and decide to grapple with the intricacies of machine code.

Andrew Hewson, well-versed in that arcane lore, reveals some of its secrets and provides a useful source of reference to would-be machine code programmers.

MANY, PERHAPS most, Sinclair users develop an interest in writing programs in machine code. The reason for that interest is not hard to trace — Sinclair Basic is relatively slow and has only a limited number of commands. You soon find that you are demanding more of your computer than its Basic interpreter can deliver and so you look to machine code to provide the speed and flexibility you require.

It is not always realised, unfortunately, that machine code programs and Basic programs are as different as chalk from cheese and so a good working knowledge of Basic is of little use in understanding machine code. We can illustrate that difference by outlining some fundamental points about machine code, beginning with three important words — **bits**, **bytes** and **addresses**. In doing so we can describe the structure of a microcomputer memory.

We then need to describe the various registers in the Z80A microprocessor, which are intimately linked to the form of the machine code instruction set. If the reader can understand the nature of the registers, he or she is half way to mastering machine code.

We also need to look at the instruction set in detail and learn a glossary of most machine code instructions. A number of short routines are included to illustrate some of the instructions.

Although the decimal version of each routine can be loaded into the Spectrum printer buffer using the decimal loader listed in table 1, it is much more satisfactory to use one of the assembler programs on the market. With such a program the assembler code version is entered from the keyboard and the program converts the codes into decimal.

Perhaps the most difficult aspect for the newcomer to machine code programming is the number of new words and concepts which must be absorbed. Let us examine one instruction as an example of what is to come. Consider the following typical compound instruction:

ld hl,(23625)

The instruction is read as **load the hl register pair** with the **bytes** held at **addresses** 23625 and 23626. Each of the words in bold print is explained in more detail later.

The instruction is conveyed in the form of three decimal numbers — 42, 73, 92. The first number means:

ld hl,()

That is load the hl register pair with the

contents of two consecutive memory addresses. The addresses in question are specified by the second and third numbers using the calculation:

lower address = first number + 256*second number

higher address = lower address + 1 or in this case:

lower address = 73 + 256*92 = 23625

higher address = 23625 + 1 = 23626

The word load is just another way of saying copy and h and l can be thought of as two special locations within the Z80A which are used for holding numbers. Thus the whole instruction means copy the contents of 23625 into register l and 23626 into register h. Notice that the lower address is the source for l and the higher address is the source for h.

The routine in table 2 makes use of this instruction to read and PRINT the value of E PPC — the system variable which holds the number of Basic line with the cursor in it. The routine works by loading the value of E PPC into the

```
10 FOR I = 23296 TO 23551
20 INPUT J
30 POKE I, J
40 PRINT I, PEEK I
50 NEXT I
```

Table 1. A simple decimal loader to place decimal machine code in the Spectrum printer buffer starting at 23296. To stop the program press STOP (Symbol shift A).

hl register pair and then transferring the values from h into the b register and from l into the c register. The routine then returns to Basic.

If the routine is called using the instruction PRINT USR 23296 command the value of E PPC will be PRINTed on the screen because the value of a USR routine is the number stored in the combined bc register pair at the completion of the routine.

Suppose, for example, the cursor points to line 10 in the Basic program. The value in E PPC at address 23625 will be 10 and this will be transferred to hl by the ld hl,(23625) instruction. The value will be transferred to bc and the routine then terminates. Hence the command:

PRINT USR 23296

will be equivalent in this case to

PRINT 10

Bits

A bit is the fundamental unit of computer memory because it can exist in only one of two states. The two states can be thought of as representing ON or OFF; TRUE or FALSE; YES or NO;

UP or DOWN; MALE of FEMALE or any other pair of logically opposite conditions. The mechanism by which a computer memory works is not really important to us but in the Spectrum the state of a bit is memorised by setting a microscopic solid-state switch either ON or OFF as appropriate.

The usual notation is to think of one state as the ZERO state and the other as the ONE state. A bit is considered to be set when it is in the state representing ONE and to be reset otherwise. That notation allows us to speak of a given bit pattern in terms of its binary equivalent and by converting the binary number to a decimal each bit pattern can be given a unique positive integer decimal number.

For example, consider eight bits of which the rightmost four are set and the four leftmost are reset. Such a bit pattern is illustrated in table 3.

The binary pattern can be converted to decimal if it is remembered that, in a binary number, the right most column is the units column; the next to the left is the twos column; the next to the left again is the fours column and so on, doubling at each move to the left. The decimal equivalent of 00001111 is therefore:

$0*128 + 0*64 + 0*32 + 0*16 + 1*8 + 1*4 + 1*2 + 1*1 = 15$

because there is one in each of the ones, twos, fours and eights columns and zero in the remainder.

It is obviously inconvenient to refer to bits as 'the rightmost' or as 'the second from the left' and so the convention is adopted of numbering the bits from the right starting at zero. It is not entirely coincidental that when that convention is used the bit number is also the number to which 2 must be raised to give the value of the column. Therefore:

bit number 2
= column value

Bit three, for example, appears in the eight column and $2^3 = 8$.

Bytes

The Z80A microprocessor which lies at the heart of the ZX Spectrum operates on eight bits at a time. The term 'operates' covers all the different tasks which are built into the instruction set such as addition, subtraction, rotation, logical AND, and so on. The form of those instructions is explained in detail later. So although a bit is the fundamental unit of computer memory, bits are usually manipulated together in groups of eight. These groups of eight bits are called a byte, pronounced bite.

Each of the bytes in the random access memory (RAM) can be used to hold a single positive whole number lying between 0 and 255 inclusive by setting or resetting the eight bits in the byte according to the binary equivalent of the number. The byte in table 3 for example, holds decimal 15.

There are 16384 bytes in the read only memory (ROM) in the ZX Spectrum and those bytes together with the electronic organisation give the computer its characters. The contents of the bytes are imprinted in the ROM when the Spectrum is manufactured and cannot subsequently be changed. It is for that reason that the memory is called read only memory — the contents can be read but they cannot be overwritten.

The unexpanded Spectrum contains a further 16384 bytes of RAM. The term random access is something of a misnomer. It does not mean that memory is used haphazardly, rather it means that any byte can be reached at any time. That facility contrasts with those of a sequential access memory like a cassette tape for which it is necessary to

move along the memory medium until the portion required is reached.

To the uninitiated, 16384 does not seem to be a convenient number of bytes to use. In fact it is a very convenient number because $2^{14} = 16384$ (as 16384 is equal to 2 multiplied by itself 14 times). In the computer world, powers of 2 are round numbers just as

Decimal	Assembler code	Comments
42 73 92	ld hl,(23625)	Load hl with the contents of E PPC.
68	ld b,h	Transfer contents of h to b.
77	ld c,l	Transfer contents of l to c.
201	ret	Return.

Table 2. A program to PRINT the number of the Basic line which currently contains the cursor. Call the routine by entering PRINT USR 23296.

millions are round numbers in everyday life. A particularly important round number is 1024 which is 2 to the power of 10. As 1024 is sufficiently close to one thousand one can justify using the letter K to represent it. (K is used for a thousand in the metric system as in kilogram — Kg, kilometer — Km and so on). Therefore, 1024 is written as 1K and 16384, which is 16×1024 , is written as 16K.

Addresses

A computer must be able to identify each location in its memory so that it may copy to and from the right location. Hence each location is given a unique **address**. An address is a positive whole number, greater than or equal to zero.

Many of the Z80A instructions are of the form 'copy the contents of the following address into such-and-such a register or register pair.' The instruction:

ld hl,(23625)

is of that form. The address following the instruction is held in two bytes. That number is the same as the number of different bit patterns which can be adopted by the 16 bits which make up

the two address bytes, so $2^{16} = 65536$. A two byte address is interpreted in the form:

address = first byte + 256*second byte
The two bytes are sometimes called the low and high bytes respectively. The two byte form of 16384 for example (the beginning of RAM in the Spectrum), is low byte = 0; high byte =

64:

$0 + 256*64 = 16384$

The Z80A Registers

A computer does not alter the contents of memory directly when it is executing a program, rather it copies the contents of a location in memory into a **register** and operates on the contents of the register. Registers have a similar function in machine language to that of variables in Basic as they are used to store numbers and can be used to control a decision. They differ from Basic variables as they are limited in number and they exist within the processor itself and not in RAM. Also they only hold one byte, or two bytes in the case of a register pair.

The Z80A is a powerful microprocessor because it has several registers and so it can hold several numbers at once. That cuts down the need to make time-consuming transfers between the processor and memory. Most of the registers have one or more special features.

The Accumulator Register

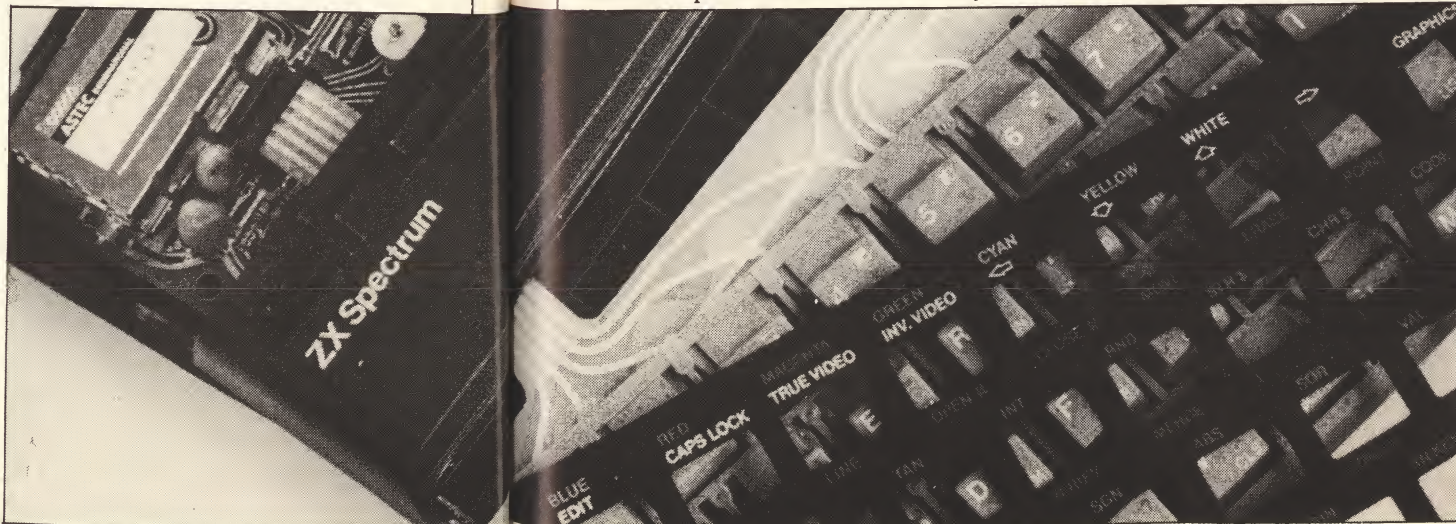
The accumulator, or a register is the most important register because most of the arithmetic instructions, addition for example, and the logical instructions operate on the contents of that register. In fact, it gains its name because the results of several successive operations accumulates in the **a** register.

Some of the instructions which refer to the accumulator use a second register or a memory address as a source of data. For example,

add a,d

instructs the processor to add the contents of the d register to the a register, leaving the result in a.

The routine in table 4 demonstrates



the 'add a,d' instruction by adding together the numbers 4 and 9, transferring the result to the c register, clearing the b register to zero and PRINTing the result. It is necessary to clear the b register because the PRINT USR command PRINTs the contents of the bc register pair in the form $256 \times b + c$. The routine does not know what was in the b register previously — it could contain any number from 0 to 255 and so it is set to zero.

The Flag Register

Most of the registers occur in pairs in the sense that some instructions operate on two registers together. The f or flag register is paired with the a register in that sense although it is limited to the push, pop and exchange instructions.

The f register is rather different from all the others because the eight individual bits in the register are used as so-called flags to record and control the sequence of program execution. Each flag is used to indicate that either one of two logically opposite events has occurred, for example the zero flag indicates whether the result of the last addition, subtraction, or whatever was

a carry or a borrow generated by bit six. It is therefore used to check if the sign bit has been corrupted. Logical instructions use the same flag to indicate the parity of the result. The parity of a binary number is the number of bits set to one. If the number is even the parity is said to be even. If it is odd, the parity is said to be odd. The flag is set if the parity of a result is even.

The effect of some instructions depends on the current setting of particular flags. For example the instruction:

jr z,d

causes the Z80A to jump over the next d instructions if the zero flag is set. If the zero flag is not set the processor executes the next instruction in sequence as usual. Thus the flag register is important because it allows the processor to make decisions and branch to another part of the program.

The Counting Registers

The b register and to some extent the c register with which it is paired is available for a number of purposes but its most important use is as a counter. We have already seen how the flow of a

the b register, by reducing the contents by one. If the result is zero then the next instruction in the sequence is executed. If the result is not zero the routine jumps d instructions. If the programmer uses a negative value for d the jump goes back earlier in the program and assuming there are no other branches, the processor will eventually encounter the same instruction again. Thus by loading the b register with a suitable value initially and setting the displacement, d, appropriately, a section of code can be executed a given number of times.

The b register holds one byte only and so it can be set to any number between 0 and a maximum of 255. Hence at most 255 passes can be made through the same section of code using that mechanism.

There are no similar facilities for making more than 255 passes through a loop, but there are a limited number of very powerful instructions which use all 16 bits of the bc register pair as a counter up to 65535. For example:

ldir

When it is encountered, the Z80A decrements bc by one; moves the contents of the address pointed to by the hl register pair to the address pointed to by the de register pair; and increments hl and de by one.

The processor repeats those actions until bc = 0. Thus this instruction can be used to move chunks of material around in memory.

The two routines listed in table 6 use ldir to copy the display file and the attributes to the top of memory (in the 48K Spectrum) and back again. The first routine sets hl to the address of the beginning of the display file and then sets de to the destination address — 58624. The bc register is then set to the number of bytes to be copied, which is

the combined length of the display file and attributable area — 6912. The ldir instruction is invoked before copying can take place.

The second routine is similar to the first except that the initial source and destination which are put into hl and de respectively are exchanged so that copying takes place from high memory to low memory.

The Address Registers

The d and e registers do not have any individual function and are mostly used as temporary, rapidly accessible memory. They may also be used together to hold the address of a location in memory which is currently of interest.

The main function of the h and l registers is together to hold the address of the instruction then uses the corresponding memory location. For that reason ix and iy are known as index registers.

The Stack Pointer

The stack is an area at or near the top of RAM which is used for the temporary storage of the contents of pairs of registers. It is designed to grow down the RAM as is filled and to shrink back up the RAM as it is emptied. The bottom of the stack is fixed and, in the ZX Spectrum, it lies immediately below the location pointed to be the RAM-TOP system variable. The top of the stack is below the bottom of the stack because it grows downwards and shrinks upwards. The address of the current location of the top of the stack is held in the sp register.

Transfers to and from the stack are made by means of push and pop instructions. For example:

push hl

causes the processor to: decrement sp;

copy the contents of h to the location pointed to by sp; decrement sp; and copy the contents of l to the location pointed to by sp.

The pop instruction is the exact reverse. In that manner the most recent pair of values pushed onto the stack are

used with a displacement, d. This means that an instruction which references (ix + d) does not use the memory location whose address is held in ix. Rather d is added to the value in ix to give a new address and originally pushed, no problems will arise.

Decimal	Assembler code	Comment
62 4	ld a,4	Load the a register with four.
22 9	ld d,9	Load the d register with nine.
130	add a,d	Add d to a and leave the result in a.
79	ld c,a	Transfer contents of a to c.
6 0	ld b,0	Load the b register with zero.
201	ret	Return.

Table 4. A program to PRINT the result of adding four and nine. Invoke by entering PRINT USR 23296.

always the values which are popped off again. That provides a simple and convenient method of storing the contents of registers temporarily, perhaps while a subroutine is called. Provided the register pairs are popped in the reverse order to that in which they were a location in memory and we have already seen how certain powerful instructions make use of hl for that purpose. Where h stands for high byte and l stands for low byte the address is held in the form:

address = $256 \times h + l$

giving a maximum of 65536 unique addresses (0 to 65535 inclusive).

The Index Registers

The ix and iy registers are each 16 bit registers and can only be used as such, in contrast to the bc, de and hl registers which we have met so far which can be used in pairs as 16 bit registers or individually as eight bit registers. The registers ix and iy are generally used in a similar fashion to the

The Program Counter

The program counter, pc, is a very important 16 bit register because it holds the address in memory of the next instruction to be executed.

The normal flow of events when an instruction is executed is as follows: 1 copy the contents of the location pointed to by pc into a special register within the processor; 2 if the instruction is held in several bytes, increment pc and copy the contents of the next location into a second special register; 3 increment pc so that it points to the next instruction to be executed; 4 execute the instruction which has just been copied in.

A jump instruction such as djnz d or jr z,d alters the normal flow of events by altering pc during step 4. Note that the alteration occurs after pc has been incremented so the value of a displacement, d should always be calculated relative to the position of the instruction following the one containing the displacement.

Flag	Mnemonic	Mnemonic Use when reset
Sign	M	P Set when the last result is negative.
Zero	Z	NZ Set when the last result is zero or a match occurred.
Carry	C	NC Set when the last result is too large to be fully recorded in one byte (or two bytes for operations on register pairs).
Parity/Overflow	PE	PO Parity — set when the last result had odd parity. Overflow — set when an operation changes bit seven as a result of an overflow from other bits.

Table 5. The four flags which control most of the operations of the Z80A.

hl register pair although the instructions which drive them require one more byte of storage compared to the equivalent hl instructions. For example:

add hl,bc

is a one byte instruction which causes the Z80A to add the contents of the hl and bc register pairs and leave the result in hl. The same instruction using ix is a two byte instruction:

add ix,bc

Another feature of ix and iy which is not available to hl is that they can be

The Exchange Registers

The Z80A possesses duplicates of each of the a,b,c,d,e,h and l registers. The duplicates are distinguished by the use of a prime; for example a', is the duplicate a register. No instructions operate on those duplicates directly but exchange instructions are available to swap two or more registers out of use and to bring their duplicates into use in their stead.

Switch Setting	Off	Off	Off	Off	On	On	On	On
Binary Pattern	0	0	0	0	1	1	1	1
Bit Number	7	6	5	4	3	2	1	0

Table 3. A group of eight bits with the leftmost four reset and the rightmost four set.

zero. Only four of the eight flags are of interest to most users. Their features are summarised in table 5.

The **Sign** flag is the simplest. By convention if a byte is being used to represent a signed number then bit seven is used to hold the sign, being set when the number is negative and reset otherwise. The sign flag reflects the sign of the last result.

The **Zero** flag is set if the result of the last operation is zero. It is also used by comparison instructions which are in effect subtraction instructions for which the result is discarded.

The **carry** flag records the overflow which occurs if the result of an addition is too large to record in the register and if a 'borrow' occurs on subtraction. There are also some rotation instructions in which the bits in a register are rotated to the left or to the right with bit seven and zero rotated to or from the carry flag.

The **parity/overflow** flag is really two flags in one. It is used as an overflow flag by arithmetic instructions to indicate if bit seven has been affected by

program can be controlled by the use of the zero flag in the jr z,d instruction. Another instruction:

djnz d

also uses the zero flag to allow loops to be constructed in machine code using b as a counter in an analogous fashion to FOR-NEXT loops in Basic.

When the instruction is encountered the Z80A decrements the contents of



Exchange instructions are executed very rapidly, much more rapidly than push and pop instructions, for instance. The contents are not copied physically from one register to the other. Rather a set of internal switches is changed so that the prime register is used by subsequent instructions and the original register becomes dormant.

Decimal	Assembler code	Comments
33 0 64	ld hl,16384	Load hl with the first source address.
17 0 229	ld de,58624	Load de with the first destination address.
1 0 27	ld bc,6912	Load bc with the number of bytes to copy.
237 176	ldir	Load, increment and repeat until bc = 0.
201	ret	Return.
33 0 229	ld hl,58624	Load hl with the first source address.
17 0 64	ld de,16384	Load de with the first destination address.
1 0 27	ld bc,6912	Load bc with the number of bytes to copy.
237 176	ldir	Load, increment and repeat until bc = 0.
201	ret	Return.

Table 6. Two routines to copy the display to high memory and to copy it back again.
Invoke the first routine by entering:
CLEAR 58623
LIST : RAND USR 23296
Invoke the second routine by entering:
CLS : RAND USR 23308

The Instruction Set

There are more than 600 elements in the Z80A instruction set as listed in appendix A. As there are only 256 different arrangements of eight bits (because $2^8 = 256$) less than half the instructions can be held in one byte. The remaining instructions are held in two or even three bytes. The first byte of a two byte instruction is either 203, 221, 237 or 253 (CB, DD, ED, or FD in hexadecimal). The first two bytes of a three byte instruction are either 221, 203, or 253, 203, (DD, CB or FD, CB in hexadecimal).

Some instructions are followed by a one byte displacement, d, or a one byte number, n, or a two byte number or address, nn, to which the instruction refers. In that way a single instruction can occupy as many as four bytes in total. For example the instruction.

jr nz,d

which we have met already requires one byte to hold the instruction itself (32 in decimal, 20 in hexadecimal) and a second byte to hold the displacement, d.

Here all instructions are referred to by their assembly language mnemonic or Op Code. The mnemonics are an abbreviated way of describing each instruction and are for human convenience only. The Spectrum will not recognise the mnemonics except through the medium of an assembler program.

Certain conventions are followed. Single registers are referred to by letter, such as b. Register pairs are named in

alphabetical order, such as bc.

A displacement, d, is taken to be positive if it lies in the range 0 to 127 and negative if it lies between 128 and 255. Larger or smaller numbers are not allowed.

The negative value is calculated by subtracting d from 256. For example the unconditional relative jump instruc-

tion:

jr d

causes a jump forward eight bytes if d = 8 and a jump backwards eight bytes if d = 248 (= 256 - 8). You must remember when calculating a displacement that a jump is made from the address of the first byte following the instruction.

The third convention is that a single byte number, n, lies in the range 0 to 255 inclusive. A two byte number or an address is represented by nn and lies in the range 0 to 65535 inclusive. The value is calculated by adding the first n to 256 times the second.

Also, nn in brackets, (nn), means the contents of the location at address nn, whereas nn without brackets means the number nn. Thus



Glossary of Machine Code Instructions

This section presents a summary of most of the Z80A instruction set. Some of the more specialised instructions for dealing with interrupts etc have been omitted.

No Operation (nop)

This is the simplest instruction and as its name implies the processor does nothing when it is encountered. It can be very useful when debugging a routine because it can be substituted temporarily for a suspect instruction without altering the functioning of the remainder of the routine. It can also be used to plug gaps introduced when making small alterations to existing programs or to cause a delay in execution particularly if it is incorporated into a suitable loop. The decimal code is 0.

Load (ld)

Load instructions are used to move one byte or two bytes between registers and between registers and memory. There are more than one hundred different load instructions which is more than any other single class. They fall into eight groups:

1. Eight bit register to register. The contents of any of the registers a, b, c, d, e, h, or l can be copied to one another.

ld hl,(23625)

means load the hl register pair with the contents of locations 23625 and 23626, whereas:

ld hl,23625

means load hl with the number 23625. Similarly (hl) means the contents of the location at the address held in hl whereas hl without brackets means the number in hl.

Finally, the destination of the results of an instruction is always given first. For example:

add a,b

means add the contents of b to the contents of a and leave the result in a.

2. Eight bit memory to register. (hl), (ix + d) or (iy + d) can be copied to any of the registers a, b, c, d, e, h or l. (bc), (de) or (nn) can be copied to a.

3. Eight bit register to memory. a, b, c, d, e, h or l can be copied to (hl), (ix + d) or (iy + d). a can be copied to (bc), (de), or (nn).

4. Eight bit register to memory immediate. An immediate is a number read from the program itself rather than from a register or from another address in memory. A number, n, can be loaded into a, b, c, d, e, h, l, (hl), (ix + d) or (iy + d).

5. 16 bit register to register. The contents of hl, ix or iy can be copied to sp.

6. 16 bit memory to register. (nn) can be copied to bc, de, hl, ix, iy or sp.

7. 16 bit register to memory. bc, de, hl, ix, iy or sp can be copied to (nn).

8. 16 bit register immediate. nn can be loaded into bc, de, hl, ix, iy or sp.

Push and Pop (push, pop)

A push instruction copies the contents of a named 16 bit register to the stack and decrements the stack pointer twice. A pop instruction does the reverse so the two instructions can be used to save register values and reload them later in the program. The register pairs af, bc, de, hl, ix and iy can each be pushed and popped.

Exchange (ex)

Exchanges can be made between hl and de, hl and (sp), ix and (sp), iy and (sp), af and af' and between bcdehl and bcdehl'. (A single instruction swaps all six of the eight bit registers.)

Eight Bit Add and Subtract (add, sub, etc)

a, b, c, d, e, h, l, (hl), n, (ix + d) and (iy + d) can be added or subtracted to or from the a register with or without the carry flag. Instructions involving the carry flag end in c.

Eight Bit And, Or and Xor (and, etc)

a, b, c, d, e, h, l, (hl), n, (ix + d) and (iy + d) can be combined with the a register using any of the three logical operators. And sets each bit in the result which was set in both sources; Or sets each bit which was set in either or both sources and Xor sets each bit which was set in one or other source but not those which were set in both.

Compare (cp)

Compare is like subtract except that only the flags and not the contents of a are affected. a, b, c, d, e, h, l, (hl), n, (ix + d) and (iy + d) can be compared with the accumulator.

Eight Bit Increment and Decrement (inc, dec)

a, b, c, d, e, h, l, (hl), (ix + d) and (iy + d) can be incremented or decremented.

16 Bit Increment and Decrement (inc, dec)

bc, de, hl, ix, iy and sp can be incremented or decremented.

16 Bit Add and Subtract (add, sub, etc)

bc, de, hl, ix can be added with or without carry or subtracted with carry only to or from hl. bc, de, sp, ix can be added without carry to ix. bc, de, sp and iy can be added without carry to iy.

Jump, Call and Return

The flag register, f, contains a carry flag, c; a parity flag, p, which is set if a result is even parity; a sign flag, s, which is set if a result is negative; an overflow flag, v, which is set on overflow; and a zero flag, z, which is set on a zero result. Those flags can be used to control jumps, subroutine calls and subroutine returns.

1. Jump (jp or jr)

The following jumps to address nn are possible: absolute jump (jp); jump on zero or not zero (jp z) and (jp nz); jump on carry or not carry (jp c and jp nc); jump on positive or negative (jp p and jp m); jump on p/v = 1 or p/v = 0 (jp pe and jp po).

The following relative jumps to an address d relative to the current position are available where d is interpreted as lying in the range -128 to 127: absolute relative jump (jr); relative jump on zero or not zero (jr z and jr nz); relative jump on carry or not carry (jr c and jr nc).

Jumps can also be made to the addresses held in hl, ix or iy (jp (hl), jp (ix), jp (iy)). The djnz instruction decrements the b register and jumps to d if b is non zero.

2. Call (call)

This instruction serves a similar function to the Basic GOSUB command. If the call condition is met then the program transfers to the instruction held in address nn. The following calls may be made: absolute call (call); call on zero or not zero (call z and call nz); call on carry or not carry (call c and call nc); call on positive or negative (call p and call m); call on p/v = 1 or p/v = 0 (call pe and call po).

3. Return (ret)

This instruction serves a similar function to the Basic RETURN command. Return conditions are available to match each call condition and returns can also be made from the interrupt and the non-maskable interrupt. (reti and retn).

Bit Instructions

The eight bits in each register are numbered from zero to seven from right to left. Each of the following operations can be performed on the a, b, c, d, e, h, l registers and on (hl), (ix + d) and (iy + d).

1. Bit Test (bit)

The bit test instruction sets the zero flag to the opposite of the setting of the named bit. Any bit can be tested.

2. Bit Set (set)

Any bit can be set.

3. Bit Reset (res)

Any bit can be reset.

4. Rotate Left (rl)

Bit seven is copied to the carry, the carry is copied to bit zero and all other bits are copied one place to the left.

5. Rotate Right (rr)

Bit zero is copied to the carry, the carry is copied to bit seven and all other bits are copied one place to the right.

6. Rotate Left Circular (rlc)

Bit seven is copied to the carry and to bit zero. All other bits are copied one place to the left.

7. Rotate Right Circular (rrc)

Bit zero is copied to the carry and to bit seven. All other bits are copied one place to the right.

8. Shift Left Arithmetic (sla)

All bits are copied one place to the left, bit seven is copied to the carry and bit zero is reset.

9. Shift Right Arithmetic (sra)

All bits are copied one place to the right, bit zero is copied to the carry and bit seven is copied to itself.

10. Shift Right Logical

As shift right arithmetic but with bit seven reset.

Rotate Left Digit (rld)

Bits zero to three of a are copied to bits zero to three of (hl); bits zero to three of (hl) are copied to bits four to seven of (hl); bits four to seven of (hl) are copied to bits zero to three of a.

Rotate Right Digit (rrd)

Bits zero to three of a are copied to bits four to seven of (hl); bits four to seven of (hl) are copied to bits zero to three of (hl); bits zero to three of (hl) are copied to bits zero to three of a.

Accumulator Operations

1. Complement a (cpl)

Every set bit of a is reset, every reset bit is set.

2. Negate a (neg)

Complement a and add one.

3. Complement carry (cpl)

Sets the carry flag if it is reset, resets it otherwise.

4. Set carry (scf)

Sets the carry flag.

5. Decimal adjust (daa)

Corrects a after bcd addition and subtraction.

Restart

Save the program counter on the stack and jump to location $8 \times n$ where n is held in the byte following.

Block Handling

These compound instructions are designed to move data or to search for data in memory.

1. Load and increment (ldi)

Move one byte from (hl) to (de). Increment hl and de and decrement bc.

2. Load, increment and repeat (ldir)

Move one byte from (hl) to (de). Increment hl and de and decrement bc. Repeat until bc = 0.

3. Load and decrement (ldd)

Move one byte from (hl) to (de) and decrement hl, de and bc.

4. Load, decrement and repeat

Move one byte from (hl) to (de) and decrement hl, de and bc. Repeat until bc = 0.

5. Compare and increment (cpi)

Compare a and (hl). Increment hl and decrement bc.

6. Compare, increment and repeat (cpir)

Compare a and (hl). Increment hl and decrement bc. Repeat until a = (hl) or bc = 0.

7. Compare and decrement (cpd)

Compare a and (hl). Decrement hl and bc.

8. Compare, decrement and repeat (cpdr)

Compare a and (hl). Decrement hl and bc. Repeat until a = (hl) or bc = 0.

**What's New For Christmas?
See For Yourself**

The Christmas ZX MICROFAIR



GO ON A SHOPPING SPREE FOR ALL YOUR
NEW SPECTRUM AND QL GOODIES!
— MANY AT SPECIAL SHOW PRICES —



TWO FANTASTIC DAYS

17th and 18th NOVEMBER 1984

AT ALEXANDRA PALACE, WOOD GREEN, LONDON N22.

ADMISSION ONLY £1.50(Adults) £1.00(Kids)

Come along on Saturday or Sunday 17th and 18th November to the Show that shows you everything for Sinclair Computer enthusiasts!

Why spend weeks looking for Christmas presents . . . one day at the Christmas ZX MICROFAIR will take care of everything!

And you'll have a fantastic day out . . . meeting friendly people . . . having fun . . . full of Christmas cheer!

Send now for reduced price advance tickets — and don't forget to note the date!

**SEND FOR REDUCED PRICE ADVANCE TICKETS
NOW AND BE FIRST IN THE QUEUE!**

Post to: Mike Johnston, ZX MICROFAIR, Dept SA, 71 Park Lane,
Tottenham, London N17 0HG. Tel: 01-801 9172.

Please send me ADVANCE TICKETS for the Special Christmas ZX MICROFAIR.

I require the following:— Adult tickets @ £1.25. Children's tickets @ £0.80p
for Saturday/Sunday (Delete as applicable)

I enclose cheque/P.O. made payable to ZX MICROFAIR for the full amount (please include cheque card
number on the back of cheques) and I also enclose a STAMPED SELF-ADDRESSED ENVELOPE for
return post.

Name:

Address:

14th
ZX MICROFAIR
ALEXANDRA PALACE,
WOOD GREEN, LONDON N22
SATURDAY AND SUNDAY
17th and 18th NOVEMBER 1984
Saturday 10.00 am to 6.00 pm
Sunday 10.00 am to 5.00 pm

EXHIBITORS!

JOIN THE RUSH . . . CALL
MIKE JOHNSTON
OR MOIRA ON
01-801 9172 FOR FULL DETAILS
OF STAND AVAILABILITY . . .
DO IT NOW!

Chapter 8

Language



With the Spectrum well established as the most popular micro in Britain, a number of languages which had their beginnings in larger computers have recently become available to Sinclair users. John Gilbert introduces the user-definable languages of Forth and Micro-Prolog and Theo Wood ventures into the educational world of Logo and the turtle.

Forth

FORTH is a new language in microcomputer circles but it has been around for several decades. It was designed originally to control a radio telescope. Since then it has been used in control applications and is seen as an ideal language for the programming of robots and industrial machinery.

Forth is now seen as an important language because it is faster in execution than Basic and it is one which can be learned very quickly by a newcomer to computing.

Some people may find it more difficult to use than Basic because it does not use line numbers and the programs written using it are in the form of several routines and not one mass structure. In that way you can have several programs in the computer at once.

One of the reasons for using Forth in the control of machinery is its speed compared to other high-level languages. For instance, Forth is more than 10 times faster than Basic and a game of Space Invaders written using it would move almost as fast as if written in machine code.

Which version?

When Forth was introduced there was no standard for the language, so an organisation called the Forth Interest Group was formed. Since then a new, second standard has been introduced which is called Forth-79. That does not differ much from Fig-Forth but Forth packages are now divided into the two standards.

Sinclair Research has just adopted Sinclair Forth for the ZX-81 and 48K Spectrum, which use the Fig-Forth standard.

It is possible to store several screens of program in both the ZX-81 and Spectrum but Sinclair Forth will store only one. If you want to use screens other than the current one you have to load them from cassette, one at a time. That is irritating because such a process is not necessary with the memory capacity available.

The other difficulty with the package is that the documentation — the two manuals — does not provide a sufficient introduction for the beginner. The difference in approach between Sinclair Basic and Forth is not made apparent enough and the transition between the



two languages is not easy using the manuals.

The second package is slightly better and contains a full version of Forth. It is from CP Software, called Spectrum Forth. Owners of 16K Spectrums will be pleased, as CP does a version of the language for that machine. Spectrum Forth also includes a program written in Forth so that the user can run an example before delving into the depths of the language.

The final version is from Melbourne House. It is also for the Spectrum but will run only in 48K. It has a bigger dictionary capacity for programs than the 16K version from CP.

It is a complete version of the language with added colour, attribute and graphics instructions which can be used to zap space invaders around the screen, if you are that way inclined. It is also possible to define your own characters just as you can in Spectrum Basic.

It is the only Spectrum package which has been endorsed by the Forth Interest Group. It is a complete Fig-Forth package in memory at the same time.

The Melbourne House manual is not good but there are several books on the market which explain Forth and its application in terms the beginner will understand.

One good book is *The Complete Forth*, by Alan Winfield. It is published by Sigma Technical Press and gives an introduction to Forth concepts. The book uses Forth-79 but there is much information which will be useful to the Fig-Forth user.

The Stack

Forth operates using an area of memory called the Stack. It can be represent-

ed as in figure one onwards. You can see that the Stack grows from one storage unit to as many as the computer can handle in its memory space. When the computer takes information off the Stack it will shrink until, at the end of the program, it will probably contain nothing.

The computer puts information, usually numbers, on to the Stack in a certain way and it is that which often causes problems for the first-time user. In figure one the computer has put the number 2 on to the Stack. In figure two another number, 3, has been added and the number 2 has been moved down to make room for the newest bit of information on the Stack.

When information is taken off the Stack, the last bit to go on will come off first. For that reason the Stack is said to be a FILO — First In Last Out — structure. Machine code programmers may find the concept familiar, as the computer also has its own internal Stack on which it stores information about the system.

In some cases if you do not take everything off the machine Stack before the end of a machine code program the machine will crash. In the same way you may get strange results if you do not control the Forth Stack properly.

As the Stack is a fundamental part of Forth and handles all the information which goes into the system it is necessary to change the way in which information is dealt with by the computer. That is especially true of the way in which Forth handles arithmetic operations.

The machine needs to know which arithmetic operation it is to perform before it gets the data on which it is to work. In the last example the data was 2 and 3 and those numbers were put on to the Stack in figures one and two. The arithmetic operator is the symbol which tells the computer what to do with the two numbers. Valid operators include *, —, ↑ but the one used in this example is the + sign. That is put on to the Stack after 2 and 3 — figure three.

Immediate mode

When the computer takes an item off the Stack it will find the + sign and know that the next items on the Stack are to be added together. The sum would normally be written as

2 + 3

but in Forth it is written as

2 3 +

The numbers are first, followed by the operator.

You can enter that sum into the computer in two ways. You could type in the following instructions

2 (ENTER)
3 (ENTER)
+ (ENTER)

The other way is to type
2 3 +

putting spaces between the 2, 3 and the + sign. Every piece of information should have a space before or after it, so that the computer can distinguish between one bit and another.

The information is then on the Stack but nothing has been done with it. It will remain there until something else is put on to the Stack or until something is taken off.

To calculate the sum of the two values put on to the Stack in figure three, the Forth print instruction can be used. That is a dot, (.). If you have set up the sum on the Stack all you have to do is to type

(ENTER)

and the answer will appear on the screen, followed by the message 'ok'.

The computer will not display the results of the calculation until you use the print symbol. You can also put the ' ' on the same line as the calculation instruction instead of typing it in separately.

The instructions used so far have had immediate results and therefore they are operating in what is termed the immediate mode.

Compile mode

An extension of that is the compile mode and it is in that where the full power of the Forth language can be realised.

If you type VLIST into the computer a block of words will scroll up the screen. The command VLIST means vocabulary list and the words displayed are those which are already known by the computer. Words such as UDG and FREE describe functions which can be performed by the computer. Those words operate in a similar way to Basic keywords but they are much more flexible and need no line number to reference them.

The word is the Basic unit of Forth programs. When a word is used the computer will perform a series of actions which are associated with it. For instance, if you type in

FREE .

the number of RAM bytes which are left free for your programs will be displayed. A beginner may find it difficult to understand the potential of

Forth word structures, as they can be used as programs.

That is not possible using the Basic language although it could be likened to calling one subroutine of Basic lines from within another.

The words which are in the dictionary, or vocabulary, when the language is initialised are small routines which have been built into the language by the original author. Usually they cannot be changed. It is possible, however, to set up your own routines and add them as words in the dictionary. That is done in the compile mode.

In the previous example the number of bytes left for programming in the basic system was determined by typing

FREE .

That produces a number with no reference to bytes. You may want to put in a printed message before or after the number, such as

NUMBER OF BYTES FREE =

That is a very simple process and you can define a word which will do it.

Definitions

First the computer needs to be told that a new word is about to be defined for the dictionary. That is done by typing a ':' — colon — at the beginning of the first line of the word. That will put the computer automatically into the compile mode and it will wait until you have typed in your word definition.

Next, and on the same line of program, you must define the name of the word. The word in this example will be called BYTES as it is best to use one which describes the operation which will be performed with its use.

The message to be displayed can be programmed by telling the computer that it has to print what follows. That is done by putting a print dot after the name of the word and then enclosing the message in quotation marks.

2

Figure 1.

3

2

Figure 2.

+

3

2

Figure 3.

That instruction is slightly unorthodox, as no space is put between the print dot and the quotation mark. Spaces, however, are put between the quotation marks and the message.

After you have typed in the message, closed off with a space and quotation mark, you must use the word FREE to calculate the space which is free in the system. Another print dot, after that word, must be used to display the result.

The computer must be told that the definition has been completed and that is done by typing a semi-colon, ';', after the definition.

As you can see from figure four, the complete program can be entered in two ways. You can either type in the word definition as one long line or you can type it in as a series of lines separated by (ENTER). The constituent parts of the definition have also been included in figure four to make it easier to understand.

There are two ways in which you can see if your new word has been defined by the computer. You could type in the word BYTES followed by the (ENTER) key and the display should then read

NUMBER OF BYTES FREE =
16512

or the number of bytes which are available to your machine.

Alternatively, you could type VLIST to look at the vocabulary. If your word has been defined correctly you should see it as the first term in the dictionary. As you can see, each new word is entered at the top of that dictionary.

You can use BYTES as a program on its own or you could use it within another word definition, as FREE was used in BYTES to calculate the number of bytes.

The Editor

The Forth Editor is one of the most important and fundamental concepts in the language. Using it you will be able to enter, amend and delete parts of Forth programs.

The Editor and the screens on which it operates are important, as they will allow you to keep a record of words which have been entered into the system. As detailed above, when a word is compiled into the Forth dictionary the constituent parts of that word are lost. You can no longer see the listing or how the word was constructed.

A permanent record of a word can, however, be kept in one of the sections of memory which are designated as

screens. Those screens will list a program just as you are able to list a program in Sinclair Basic. That is called the source, or text, listing.

Melbourne House Forth can store up to 12 screens of words. They are numbered from zero to 11. Each of those screens takes 1,024 bytes of RAM and is structured as 16 lines with a potential of 64 characters on a line. That has nothing to do with the lines on the screen of the Spectrum or, if you are using it, the ZX-81, but it is just a method of dividing sections of memory in the machine.

The screens take 11K of RAM and, because of the speed at which information can be accessed by the Forth system, Melbourne House has called this memory the RAM disc. When you first enter the Forth compiler you should type

DISC—INIT

to clear the disc and leave the RAM ready for new input.

You should then type
EDITOR

to invoke the commands for the source input and line manipulation. To select the screen page which you want to use, type the page number and CLEAR. In this case we shall use page one.

1 CLEAR

Then list that page by typing 'L'. You will see a heading at the top of the page which will indicate that you have accessed the correct screen. You will then see a list of numbers running down the left-hand side of the page.

It is best to start putting your word definitions on page one as page zero is reserved for comments about your program code. That means you can document your Forth words on that page and the explanations can be assessed by typing

0 LIST

If you want to remove all the source from that or any screen type

0 CLEAR

where '0' is the number of the page.

To put textual comments on page zero or any other screen you should type the line number, followed by the letter 'P' and then the remark. That type of statement will operate in the same way as REMARK in Sinclair Basic — in other words it will have no effect on the running of a Forth program.

Just as screen zero can be used for comments, so can line zero of every screen page. That means you can put a title on a page of words. That is done in the same way as it was for screen zero:

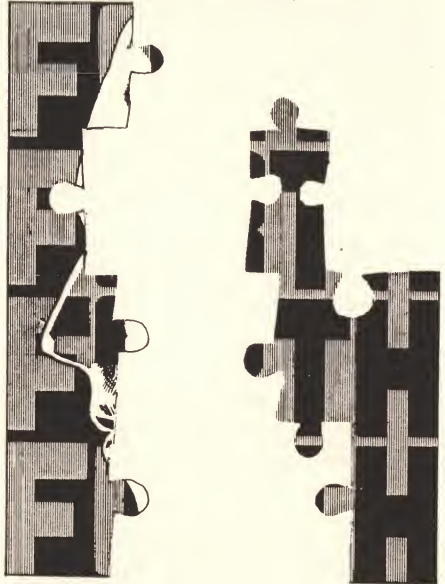
1 P THIS IS THE TITLE

To put a word definition on to the

screen you must select your page, which is one in this case, and develop the word definition using the colon (:) to start and the semi-colon (;) to finish in the usual way. The only difference is that you have to add a line to the beginning of the text.

To put the BYTES FREE definition on to screen one you would type
1; BYTES "NUMBER OF
BYTES FREE=" FREE ;

If you have any extra code you could put it on to other lines of the screen, although 64 characters on a line should be sufficient for most purposes.



SAVEing

You can put several words on one screen. All you have to do is to indicate to the computer that the text you are entering is a definition by using the colon at the beginning of the word.

After you have entered the definition you may want to save it as a source listing before you compile it into the Forth dictionary. To do so you must save the whole disc file onto tape, using the SAVE command. To check that the file has been saved correctly most Forth packages have a VERIFY command.

When you have SAVED the disc source code you can safely compile the original, which is still in memory. To do so type the colon which puts the computer into compile mode. That will compile all the words which are contained on the disc. If you want to compile a selected word you should type the colon, followed by the name of the word.

To check that the definition has been compiled correctly VLIST the present dictionary. Your new definition should be the first on the list.

Try running your new BYTES word. If it does not work, you have the source

listing on tape which you can alter and compile again. Just LOADT the relevant disc into memory and type the number of the page on which you put the word definition, followed by the usual space and the command LOAD.

You will then have the relevant page in memory and all you have to do is LIST it and make your alterations by invoking the EDITOR for your particular system.

The Forth Editor provides some complex commands which will operate on screen lines and words. Some versions of Forth may have slightly different commands, so you should refer to your manual before making alterations.

Keywords

When you can save words on to disc and recall them you can develop your own programs. To give an introduction to the more complex applications figure five will show what can be done with one Forth word.

You may have noticed when looking through the main dictionary that few mathematical functions are included. One of the reasons is that Forth is adaptable and, unlike many versions of Basic, you can define your keyword definitions.

In mathematical circles the function for finding the square of a number is important and therefore useful to have in your Forth dictionary. This word definition will put it in there for you:

:SQUARE DUP DUP * ;

DUP means double. The action of DUP puts a specified value on the Forth stack and then generates a duplicate of that number which is put on the stack on top of the first value. You can see the action of this word by typing:

4 DUP .

in immediate mode.

```
: BYTES
"NUMBER OF BYTES FREE="
FREE ;
(a) word definitions on several lines

: BYTES "NUMBER OF BYTES
FREE="
FREE .
(b) word definition entered as one line

:/BYTES"/NUMBER OF BYTES
FREE=/
"/FREE /
(c) break-down of definition separated by '/'
```

Figure 4.

Now you can understand how DUP works you should be able to see how the definition SQUARE operates. It puts four copies of the specified number on to the stack and then it does the multiplication using those values. The result will be the square of the original number followed by that root number. For instance, if you typed

100 SQUARE

you should finish with the computer output

100 100 ok

You could produce word definitions which will give you a full set of mathematical functions and even give those new words the names of the keywords which provide the same functions in Basic.

Graphics

The graphics commands which are used in the Sinclair and Melbourne House Forth packages are not part of the original Fig-Forth standard. The commands, which allow the user to manipulate attributes, draw lines and produce user-defined graphics in Forth programs have been added to make the language more attractive to a market which, at the moment, is still engrossed in obtaining better and faster graphics effects.

The most basic commands are the colour and attribute functions. In Basic, the command to change the screen colour is:

PAPER 1:CLS

In Forth the word to change the paper colour of the screen is slightly more involved. The Forth command to change paper is like any other command in that language. We have to take the stack into account and supply it with the number of colour we want. To change paper we can type:

1 PAPER

You will have noticed that something strange has happened. Every time you type a letter on the screen the attribute containing the current colour is set to one, or blue, and the character square you have just used turns to blue. If we want to change the whole screen to blue, we have to resort to our knowledge of Basic and type:

1 PAPER CLS

You should then have all the screen, except for the border area, covered in blue. The INK will be black but can be changed by typing the ink colour and the command:

6 INK

It should be clear by now that you can manipulate the attribute file of the

Spectrum more easily in Forth than in Basic. You should have control over either the full screen colour or a specific character block through just two commands in Forth. The same would take a PRINT AT instruction followed by a PAPER and colour number or CHR\$ in Basic.

To take an example, we could change the universal paper colour to blue with:

1 PAPER CLS

The ink can be set using:

6 INK

and the local character screen colour can be re-set using:

3 PAPER

```
INIT-DISC
EDITOR
1 CLEAR
0 P TITLE OF WORD

1: BYTES "NUMBER OF BYTES
FREE=" FREE;

SAVET
VERIFY
:
```

Figure 5
Step-by-step entry of a word on to a screen.

The border can also be changed using the same format:

4 BORDER

You can see that those commands are similar to Basic when writing programs in Forth and very easy to use. One useful word which we would develop with the aid of them is an extended CLS command. At the moment CLS will clear the screen and re-set the cursor to the home position at the top left of the screen. The only difficulty is that when a program has finished, or you have changed the colour of the PAPER, that old paper colour will remain on the screen. I will call our new word RE-SET.

Type in the word in figure six and to test it, type:

1 PAPER CLS

4 INK

That should make the screen almost unreadable. Type the new word, RE-SET, and the screen should clear. The alternative is:

7 PAPER 0 INK CLS

and, as you can see, there is a considerable amount of saving of time and finger work.

PLOT and DRAW

Attribute manipulation is the easy

part of graphics using Forth. The other two parts are plotting which, in turn, leads to line drawing. The commands to do both those tasks are again very close to Basic.

The PLOT instruction will do two things — position the cursor at a specified x, y set of co-ordinates on the high-resolution screen, and fill the memory location dot on the high-resolution screen with the current ink colour.

As an example we could plot a red dot at the bottom left-hand corner of the screen. To do that type:

CLS 2 INK 0 0 PLOT

We could draw a line using the PLOT word and a loop command such as DO, which is explained in both the Sinclair and Melbourne House Forth manuals. That routine will draw a line in about the same time as a line would be drawn in Basic, because the Forth language is faster than Basic.

It is not necessary to use PLOT to draw lines as there is a DRAW word available in the current Forth dictionary. That functions in the same way as PLOT but you have to put a PLOT command before it to state the point of origin of the line.

Unlike the DRAW command in Basic, the Forth command will not take into account any error messages if the line should go off the screen. That may sound ideal — no error message when something goes wrong — but then you must remember that the DRAW command will start to draw lines which are off the screen and when you DRAW again, nothing will appear.

Figure seven is a Forth word, SQUARE, which will draw a square at the bottom left corner of the screen. It shows that the screen co-ordinates used in the DRAW command are absolute and in that way the command is different from that in Basic.

It is a good idea, as suggested in the manual, to try drawing a square in Forth and then computing the speed at which it is drawn in Basic. You should notice a considerable difference, Forth being the faster of the two languages.

The other commands which can be used in Forth also relate to Basic. They are BRIGHT, INVERSE, FLASH and COVER. All have their arguments before the command and the arguments can be 0 or 1, just as they are in Basic.

Sound

Music is produced using the BLEEP command, which functions just like the Basic BEEP command except that the argument, duration and pitch relate to

machine code cycles and, as a result, one cycle will be performed much quicker in Forth than in Basic. For that reason the numbers used in the word definition are longer than are used with Basic.

Type in figure seven and then type figure eight, which is the new musical command called MUSIC. That will produce a series of BEEPs after the square has been produced on the screen, although it will seem as if both actions have taken place together. To obtain that effect type in:

SQUARE MUSIC

The lower the arguments of BLEEP become the higher the pitch and the lower the duration will be. The highest of either number you can reasonably expect to hear is:

90 90 BLEEP

That is extremely quiet and to gain the full effects of Forth music you should use an amplifier. Because of the quality of sound, relative to Basic, which BLEEP provides it is worth playing with that command.

Variables

It is not necessary to know much about machine code to understand what is happening in the Forth stack when a variable is created but if you can follow the arguments about numbers you should have a better idea of what is happening.

The POKE statement in Basic stores a 16-bit number in a memory location. An 8-bit number can be anything from 0 to 225 in decimal or 00 to FF in hexadecimal. A 16-bit number is a multiple of the first 8-bit number. It can range from 0000 to FFFF in hexadecimal.

To enter a 16-bit number on to the Forth stack as a variable we have to use a Forth dictionary word to define it. The word is:

VARIABLE

and the value of the variable goes before it and its name follows it.

To create a variable called VARS and enter the number 50 the following line could be used:

50 VARIABLE VARS

There must be a space between the 50 and VARIABLE and between VARIABLE and VARS.

If you press ENTER, your variable will be stored by the system. The value is not stored necessarily on the stack but the memory address at which it is stored goes on to it. When you ask for the value of the variable you have created, its address is on the stack and can be

used as an index to the value which is being stored elsewhere.

It is similar to the way in which Basic variables seem to operate for the user where the variable name, in this case VARS, is used as an index to represent a value, which is 50 in this case.

To get back the value of the variable we will need a new Forth symbol which means, for our purposes, Fetch. It is symbolised with an '@' sign. To get back the 50, using the VARS name as an index, all you have to type is the variable name, together with Fetch and the printing dot. All three symbols have spaces between them:

VARS @

The screen display should then include

50 ok

As an extension of the variable technique, it is possible to create what in Basic would be called an array. For those with little knowledge of that type of data structure, an array is a table or list of values which are indexed, or named using one overall label and a distinct index number for each of the



elements. For instance, we could give our example array the name VALUES, tell the computer it contains 10 elements, and then number each of them from one to 10.

First we would need to clear space in the Forth dictionary, into which we will put our values. When a numeric array is dimensioned in Basic, the system gives each element the value of zero. For our purposes and, just to be awkward, we will give our array the initial value of one.

To do so we will have to use another new Forth word called ALLOT which has the function of giving the specified number of elements to our array. The

Figure 6.

```
: RESET PAPER 7 CLS
INK 0 ;
```

Figure 7.

```
: SQUARE
0 0 PLOT 0 50 DRAW
50 50 DRAW
50 0 DRAW
0 0 DRAW ;
```

line to dimension the array is:

```
1 VARIABLE VALUE 10 ALLOT
That should be followed by ENTER.
```

Taking the line apart, the number '1' is the initial value given to the array; VARIABLE will allocate a variable space for the array; VALUE is the number of elements in it.

Each of the elements is one byte long and so it is not possible to hold one number in each one. We have to pair those 8-bit bytes into 16-bit words to store our 16-bit values. That means we can store up to six values in our array.

If you thought that creating a variable was difficult and you did not like the idea of indexing and machine code addresses, you will be pleased to learn that there is another way of creating an integer variable, or one which does not contain a decimal point. Using the following technique, the computer will put the value of the variable and not its storage address on to the stack.

As with all Forth words, the value of our variable is put first on to the defining line. That is followed by the dictionary word, CONSTANT. This word tells the computer that you want to assign a value which has been put on to the stack with a label, just as would happen in Basic. Following the definition we must put the name of the label, which we will call DEF:

50 CONSTANT DEF

After you have pressed ENTER you can refer to that value on the stack by the name DEF.

Data entry

So far we have dealt only with entering numbers on to the Forth stack. We have not attacked the entry of strings because it is slightly more difficult than typing in a figure which the computer will then put onto the top of the stack. It is possible, however, to implement a simple utility which could be used for entering data into a system such as the one used to store patient records in a hospital.

To produce that on-screen user utility program we must first enter two Forth words into the dictionary. I shall use the Skywave Software's ZX-81

Forth in this example although, as it provides fairly standard Fig-Forth, people with Sinclair and Melbourne House Forth should be able to do the same on their computers. It is slightly easier, however, to illustrate the point using the on-board ROM of the ZX-81 Forth package.

Figure nine provides a simple Basic-type INPUT routine which will put a question mark on to the screen and expect you to enter a one-word response. Typing ENTER or BREAK after the data will put the string on to the character stack.

That stack is like the one we used for entering numbers and will accept one character at a time. Each character is converted to its ASCII code, which again is a number. The character stack behaves in a similar way to the one we have been using previously, except that it is for the exclusive use of symbols such as a, b, c.

INPUT is the name I have chosen for the new dictionary word which will act as a bridge between the user and the character stack. The first thing it will

Figure 8.

```
: MUSIC
400 400 BLEEP
300 300 BLEEP
200 200 BLEEP ;
```

do is to generate a carriage return, or new line, using CR. The question mark, which acts as a prompt for the user, can then be displayed on the screen using the print command, '.', followed by the explanation string announcing the program which is in the usual quotation marks.

The next part of the new word is S@, which may look strange to you. If, however, you take the two symbols apart and analyse them you can see what this operator does. Look in your user manual and you will find that wherever the '@' symbol is used as a pre-fix or suffix the Forth word to which it is attached will do something with machine addresses.

The 'S' symbol can be taken as meaning string, so 'S@' will provide a link between a string of characters on the stack and the address of those characters. That gives access to the string when you want to call it back off the stack and use it on the screen.

Following the 'S@' symbol is another carriage return and the end of word marker.

Data display

That is satisfactory for entering words but you will also need to get the string you want off the character stack and on to the screen. This can be done with the word defined in figure ten.

The DISPlay word will output the text on top of the character stack and then return the system to the user. A carriage return prepares the computer for the output of text on a new line, although that is unnecessary, as in most cases an automatic return will be made. Again the next command can be deciphered by taking it to pieces. The CO goes back to the days of mainframe computers with their operator consoles. It means the device on which the text to be output is printed. In this case the screen of a television set is used. The 'P' means print and W stands for 'word'. With a little thought most Forth words can be deciphered in that way.

The two words mentioned can be used in their own right to make input and output easier for the user but just to show how they can be incorporated into other programs, figure 11 provides an EXAMPLE.

The screen is cleared, using CLS, and a prompt is given to the user. On the next line of the definition is the INPUT word, which will produce the question mark, to which the operator must respond. Once an entry has been made, remembering that it must be one word in this instance, the screen is cleared again and the string is quickly taken off the character stack ready for display on the screen. Type all three definitions into the computer, produce a permanent version on tape, and you will have a routine which mimics the Basic INPUT and PRINT routines.

Character stack

To get the feel of what is happening inside the computer we must see what is happening on the parameter or number stack which we have been using until now and the character stack to which you have just been introduced.

In ZX-81 Forth the string is stored on the character stack and a reference, or index, to its whereabouts is put on to the parameter stack. As more strings are put on to that stack the parameter stack also grows with the index addresses of each character string.

If we were intending to take an element off the parameter stack we would use the command DROP followed by

```
: INPUT CR ." ? " S@ CR ;
Figure 9.
```

the '.' symbol if we wanted to print it. A similar command is used to take a figure, such as 'A', off the character stack, but this time we put a 'C' before the DROP. In that way you can enter a string but take just one of its characters off the stack at a time.

The number and character routines can be used for all types of data entry to the Forth system. Number entry is slightly easier but with some simple additions to the Forth dictionary we can make it behave like Basic, at least to the unsuspecting novice user.

Human interface

The human interface is the part of a computer or a piece of software which takes the human into account during its operations.

Schools and colleges are the obvious places to start as it is at those institutions that computer studies and, for the first time, Forth is being taught. It is at school that most young people learn about computers and examination boards are now treating Forth as a

```
: DISP CR CO .W CR ;
Figure 10.
```

second language to Basic and Fortran.

There are two levels at which it can be used at school. The first is with the teacher, who can add words to the language to demonstrate the principles of a computer in operation. Forth can be turned into an even more high-level language by adding better input and output interfaces, so that the student can use it with ease. They can be stripped from the language as a pupil progresses. Eventually the pupil will be able to use Forth at assembly language level.

Forth will be of use to the pupil on that level in education as a second language, and it is finding a place alongside the traditional and old school languages such as Basic and Fortran. It will train the pupil in machine concepts and make it easier to understand machine code in the long run.

Another area which can benefit from the processing power of the language is mathematics, not so much in education but in industry and the sciences.

Floating point

Unfortunately most of the packages on the market for the Spectrum and ZX-81 offer only integer representation, such as 1324, and not floating point representation, such as 1.324. One company which provides floating point is CP Software, which produces 16K and

48K Spectrum versions of Forth.

Integer Forth is faster than floating point but is usable for most purposes. Floating point Forth lays the emphasis on mathematical procedures and is ideal for scientific work in, say, a laboratory. Such a package could even be used to check variables in an experiment.

The level of sophistication of the language in the laboratory environment can be controlled by defining new words to cope with the expertise, or lack of it, of the staff. One scientist may be able to use Forth unaided whereas another may need the help of an easier Forth structure. Whatever the needs, the Forth language will be able to cope in most situations.

Forth is also ideal for mathematicians as they will be able to cope with the Reversed Polish Notation which is the curse of any student learning the language. The problem is that the arithmetic operators, such as + or -, are put after the operands, such as 2 3 + which means 2+3. Mathematicians can handle that easily and it is one of the factors which increases the speed at which Forth runs, as the computer also handles formulae like this easily.

Unfortunately the graphics on all the Spectrum packages are only elementary but they show the speed with which pictures can be drawn and manipulated on the screen. The inherent speed factor means that design programs can be written in the language for use by engineers, or even craftsmen who work in metals or pottery. On a more educational level, Forth could be equipped with turtle graphics commands, which means that students can interact with the computer to produce and change pictures on the screen.

As Forth can be compiled, graphics routines using the language can be produced quickly and attached to arcade games or even adventure games.

System design

One area in which Forth can be used to great effect but which is not thought of in Sinclair circles is its ability as a software systems design language. Usually the design of languages and other utility packages for microcomputers and big mainframes is done using either assembly language or specialised systems design languages such as 'C' or machine code.

Again the compilation facilities of Forth make it an ideal language with which to produce other compiled or assembled languages. Utilities such as assemblers could also be created in

Forth, which would make it a tool for the design of software which would eventually be used in another micro-computer.

At the moment Forth is being used to write computer language interpreters for mainframe computers but such a system could be used by software houses in the Sinclair market. It would also be possible to write software using Forth on a larger development system, such as those used by Psion, and then download the software to run on a Spectrum or ZX-81.

The advantage of using Forth for software design is that it is simpler than assembly language and much easier to use than machine code. Large software projects, such as the design of new languages, can be done quickly using Forth and the resulting listings of such projects will be easier to read.

The other advantage is that people who do not know enough about machine code to do complex operations on the Spectrum can produce programs with the power of the language which look professional.

Robotics

The most obvious application in industry is the control of robots in a factory. Forth can be adapted to fit any particular input of output needs required, so that the people who work with the machines can program them if a change of task or speed is required. Large-scale changes can also be incorporated into operating program design in a faster time than if machine code was being used and the machines, which cost thousands of pounds an hour to maintain, will not be idle for long.

The input/output facilities of Forth can also be a benefit in the factory

environment which may often have to be under strict control. For instance, a computer programmed with Forth could monitor the temperature on the factory floor by taking readings from a thermometer and then adjust the thermostatic controls. In a similar way various parts of a production line can be controlled very accurately.

```
: EXAMPLE CLS
" USER UTILITY PROGRAM "
INPUT
CLS
DISP ;
Figure 11.
```

All of those uses may seem a long way from the Spectrum and the humble ZX-81 but most of the uses could be programmed on to a Sinclair machine, especially if it contained a Forth ROM. Forth will make even the ZX-81 into a powerful machine.

Further reading

Those are just some of the ways in which Forth will help the human user get along with computers. As an adaptable language it should be useful for most purposes and most users but all the Spectrum and ZX-81 packages have weak areas which could be overcome with a little more thought.

None of the packages on the market is helpful where input/output is concerned. Nobody has gone to the trouble of producing worthwhile examples to drive any peripherals in the hardware line where Forth comes into its own.

One other blasting criticism is that no companies have backed their Forth interpreters with software which can be run on the Spectrum or ZX-81. For instance, the least that could be done are some utilities or programming tools which would make life easier.

In the meantime, you may like to look at some recommended books on the subject. *The Complete Forth*, by Alan Winfield, is published by Sigma Technical Press. It will introduce you to most of the concepts of Forth but describes the Forth 79 standard which is slightly different from Fig-Forth. Others books which can be recommended are best obtained from a good library. They include *The Systems Guide to Fig-Forth* and *Starring Forth* by Leo Brodie.

Alternatively you could join the Forth Interest Group, an organisation which started in the U.S. but which has a branch in Britain. It can be reached at 15, St. Albans Mansion, Kensington Court Place, London W8 5QM.



Micro-Prolog

THE MICRO-PROLOG language package from Sinclair Research arrived at an appropriate time to give a new direction to the software market.

The term Micro-Prolog is short for microcomputer programming in logic and the language is one of a new generation which could be termed, under certain circumstances, artificially intelligent. That means that the computer will appear to give intelligent answers to questions which it is asked.

From that description of intelligence you should be able to see that Micro-prolog is nothing like Basic. In Basic you would write a series of commands which you would then expect the computer to execute in a sequence. Micro-Prolog, however, expects a series of rules, rather like those built into a database. The rules can relate to each other or be entirely separate.

The language will provide answers to any questions you might pose to the computer which relate to the rules you have entered into the language database. Even though it is capable of artificial intelligence, the language will not be able to answer your questions if it has no rule to deal with them. Neither can it make assumptions or draw conclusions without those rules.

Simple learning

To aid the construction of your programs there are several utilities in the package. One is called Simple and it will convert the base language into a language which a beginner can use. It does so by adding command words to the database which will make the task of programming easier for the user.

Prolog programs are loaded in a different way from those written in Basic on the Spectrum. The computer will load software in blocks and the block numbers will be displayed for each, along with the message 'BLOCK OK' or a read error which will break the loading process.

Once Simple has been entered you will be ready to write some rules. You could, for instance, build some rules about a family and the relationships in it.

The family could include Clive, Mary, Robert and Rebecca. If the computer is to answer questions about those people it must first have a set of rules to which it can refer. They would include

Robert is the brother of Mary, Clive and Rebecca; and Mary is the sister of Clive, Robert and Rebecca. Those rules would be entered for each of the family members.

The rules have to be converted to a form of understanding of the language. Robert is the brother of Clive would be translated into add (Robert-brother of Clive). The middle name is the relationship between Robert and Clive and must consist of one word, hence the hyphen.

It is also possible to qualify a data name such as that of a member of a family with several tags which would identify it further when the database is interrogated. To do so we can use a statement which consists of one data name and one relation name. You may want the language to know that Robert is male or that Rebecca is female. To do that you would type add (Robert male) and add (Rebecca female).

True or false

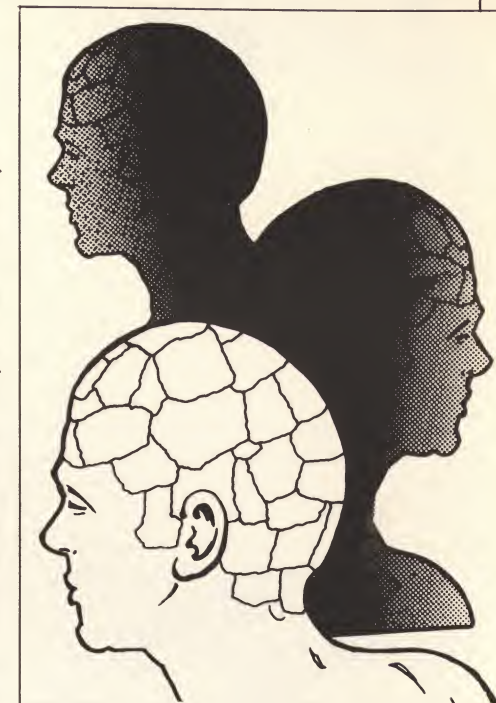
To query the database you can use several commands, including 'is' and 'which'. The 'is' command is used to test if a statement is true or false. We could ask the question 'Is Mary the sister of Rebecca?' The computer could then give a YES or NO answer. It will give a NO answer in two cases — if the statement is false or if the statement rule does not exist in the database.

The other query command is 'which' and that is more flexible. It will allow you to find the relationship between one item in the database and another. With the 'which' command you need to know only one of the items. For instance, if you want to know who is Rebecca's sister you would use the form which (x:x sister-of Rebecca). The x specifies the unknown quantity, as in algebra, and in this case it refers to a sister Rebecca might have.

When that command is entered you should have the name of Rebecca's sister, followed by the statement 'no more answers'. That means that, using the rules it has obtained from you, the computer has found the corresponding relations in the 'sister-of' category.

Useful database

Data structures such as those can be built with any kind of relationships. To fit Micro-Prolog into the definition of a language you have to stretch that defi-



nition slightly to include databases and programs which store lists of data which can be accessed depending on the user's needs. The difference is that Micro-Prolog is more adaptable to the needs of the user than a normal database written in Basic or machine code.

Although the language seems to be specialised it has several uses, two of which are the creation of databases which can be used by almost anybody and also the creation of educational teaching tools.

In the utilities line the Prolog language has been used to build a model of the London Underground system. The various stations in the network were fed into the computer and the best, and quickest, possible routes between stations were found.

Micro-Prolog can also be useful in the classroom. The database can be programmed with information about a subject and the student can interrogate the database and not have to think about what is happening in the computer.

The final test of software is whether it improves or advances the use of a computer and this package certainly does so. The only criticism is its price, £24.95, which will deter some people buying what is an example of good software.

Despite the price, Micro-Prolog will prove that the Spectrum can be used for serious purposes and that the computer has a good future in schools and other institutions.

Sinclair Research, Stanhope Road, Camberley, Surrey GU15 3PS.

Logo

THE ARRIVAL of the Sinclair version of **Logo** must be considered as the important event in the year as regards learning software.

Logo is the high level programming language developed at MIT — Massachusetts Institute of Technology — by Seymour Papert and his associates. In his book *Mindstorms* Papert outlined the reasons for the importance of Logo; it provides an introduction to computing for young people in that by controlling the micro-world of the turtle they become actively involved in computing processes. Moreover, mathematical



ideas, especially geometry, are made concrete rather than abstract. Logo can also involve those whose background is in the humanities and bring them into contact with a technology that would otherwise seem strange and alien.

Sinclair Logo has been developed by the same team, Logo Computer Systems Inc., who developed the original Apple version as well as the Atari and IBM PC versions and, later this year, the implementation for the BBC micro. The Sinclair version includes a cassette tape with the program and two ring-bound manuals.

Logo I covers the use of turtle graphics and Logo II is a programmer's reference manual. There is also a Logo ready reference card. Logo is considered important in that, unlike Basic, it reinforces good programming techniques, encouraging the user to build a program through procedures rather than a spaghetti of GOTOs. That means a program can be structured by breaking each part into its smaller components before incorporating those into the larger structure of the complete program,

reflecting the methods used in programming in the commercial world.

Once **LOADed**, Logo operates in three modes: Logo mode, TO mode and editing mode. In Logo mode any procedure or command such as **BK 10**, **FD 10** — Back, Forward — will be executed on the screen immediately after **ENTER** has been pressed. That means that at a very early stage you can see the movements resulting from commands immediately and evaluate their success. There are two lines for commands and 22 lines for graphics.

The screen turtle takes the form of a triangle which shows the direction of the heading, although **HT** — Hide Turtle — will increase the speed at which the turtle can move around. Each procedure has to be started by the use of **TO** and the title of the procedure and, after this, each entry will be prompted by a **>** instead of the question mark for the Logo mode. Commands can be **ENTERed** until the end of the procedure, which is signified by **END**. The message will then come on the screen — for example, **SQUARE** defined. It is now possible to use **SQUARE** as a command in Logo mode.

The **EDIT** mode is entered by using the message **ED** followed by the name of the procedure to be edited. The editor is powerful; not only can the cursor be moved in all four directions by use of the cursor keys but there are many other useful features such as **E MODE** — extended mode — 5 which moves the cursor to the beginning of a line and **E MODE E** which moves the cursor to the end of the text. In that way the editing mode is rather like a word processing package which allows swift movement over a piece of text for correction purposes.

One factor which has importance is the memory space available for utilising Logo features. Logo sits between the addresses 24832 and 65024, its workspace comprising 2293 nodes each of 5 bytes. The command **NODES** will return the number of free nodes in the workspace. A simple **SQUARE** procedure — **TO SQUARE, REPEAT 4 [FD 30 RT 90], END**, requires 120 nodes, which means the workspace can cope with a large number of squares. Any procedures can be **SAVED**, for later use and development, to either cassette or microdrive.

Just as the editing mode is relatively simple, so the error messages are easy to understand. 'I don't know how to...' will appear as an error message if the user **ENTERs** a named procedure in

Logo mode which has not been defined, and other error messages include 'Not enough inputs to ...' and 'Turtle out of field'.

Logo is a high-level computer language in that its commands are near to language as it is spoken in a shortened form. A list of some of the words used will show how simple the actual vocabulary of Logo is: **SETPC** — set pen colour followed by a Spectrum colour number; **PD** — pen down; **ST** — show turtle. The similarity to spoken language is the main appeal of Logo, as it makes the language more accessible.

That is not, however, the whole story because although the vocabulary is easy the grammar is difficult. It is necessary to observe certain rules in the use of the vocabulary, spaces have to be put in the appropriate places and square brackets have to be used in some circumstances. That means that there still is some learning to be done and attention must be paid to the correct entry of commands; it is no good typing **BK30** as the error message will appear: 'I don't know how to **BK30**'. There must be a space between **BK** and **30**.

With graphics Logo does, on the whole, satisfy the criteria stipulated by Papert; it is a tool by which you can explore the world of geometry and learn by doing rather than by being told. It provides an entrance into the world of computer programming and inculcates good programming practice.

It is a mistake, however, to think of Logo as merely a learning tool for geometry. Sinclair Logo is the full implementation of the language, providing powerful list processing facilities. A list can be a group of words, other lists or both; for example [apple pear orange banana plum] is a list and a number of commands can be used to manipulate the words contained within the list. **PR**



— print — **ITEM 3** [apple pear orange banana plum] will return orange, and **PR LAST** will return plum.

Logo has arithmetic functions which can operate in the infix form, where the signs **+ - * /** are placed between the numbers, thus **PR 6 + 789** returns 795. The prefix form also can be used for addition, division and multiplication where the words **SUM**, **DIV** or **PRODUCT** are placed before the two inputs, thus **PR SUM 8 9** returns 17.

Logo is likely to be used in control situations without using machine code. Not only can a robotic device be controlled via the commands **STARTROBOT** and **STOPROBOT**, which causes the commands **FD**, **BK** and so on, to be executed by the device, but there is also the facility to monitor inputs and outputs. **SERIALIN** will read everything that arrives at the serial port and **SERIALOUT** will send a byte to the serial port — **RS232** interface.

Logo 2, the programmer's reference guide, is precisely what its title implies and it would seem that there will be a whole industry of new books based on interpreting and explaining the concepts contained in it. That would be a pity as, ideally, the Logo world should be explored without a phrase book; it is the process of discovery which is important rather than the end product.

Chapter Seven, Conditional expressions and flow of control, illustrates the nature of the manual. Anyone who has learned some Basic has probably learned the **IF** command: **IF a=5 THEN GOTO**. The similar command in Logo is explained in the manual in the following way: **IF pred instruction-**

list1 instructionlist2. It is necessary to enclose the instruction lists in square brackets. That illustrates that it is misleading to think of Logo as always being more simple than Basic.

One notable feature missing from this version of Logo is the absence of the **SPRITE** function which both the Atari and the forthcoming BBC version have. That allows the user to piggyback a graphic onto a sprite and then set the speed. As well as having colour and sound the dimension of movement can be easily added, obviously an extra, exciting attraction. Clever programmers will probably use the facility to **LOAD** a machine code routine into the Logo workspace to overcome that lack but it makes the process more complicated. There is no **blockfill** command either.

Overall, Logo is the package which could become the programming language used throughout the school system and, when the examination boards recognise its use, there will be the extra motivation to buy it. Added to that is the fact that control packages are being developed by the Microelectronics Education Programme and the Open Uni-

versity which indicates that, as well as being the means by which users can explore geometric and mathematical concepts in an interactive way, they will also be able to control and manipulate electronic devices.

The package costs £39.95 and consequently there is likely to be some price sensitivity to Logo in the home market, especially when considering the bottom line value of the medium — the cassette — on which it is produced.

Logo fully illustrates Papert's insistence that the emphasis should be on user control: forget about zapping answers in response to sums, control what is happening on the screen and at the end of the cable instead. Problem-solving becomes more interesting when the results can be seen immediately and are more spectacular, without the need for complicated Basic commands. Above all, once learned, Logo is a gateway to the complex world of mathematics and computer control.

LOGO
Memory: 48K
Price: £39.95
Gilbert Factor: 9

Why wait any longer?

Upgrade your

16K
ZX SPECTRUM
Now!

The CHEETAH 32K RAMPACK simply plugs into the user port at the rear of your computer and increases the memory instantly to 48K.

- ★ Fully compatible with all SINCLAIR accessories via rear edge connector
- ★ NO NEED TO OPEN COMPUTER AND INVALIDATE GUARANTEE
- ★ Why send your computer away and wait weeks for upgrade
- ★ Fully cased tested and guaranteed.

only £39.95
including VAT and P&P.

Now make your Spectrum and ZX-81 Talk

Compatible with Interface I & II

The Cheetah "SWEET TALKER" just plugs into the back of the computer using the existing power supply. Based on an allophone system you can easily program any word sentence or phrase. Fully cased, tested guaranteed and compatible with all SINCLAIR accessories via rear edge connector. Complete with demonstration cassette and full instructions. No more lonely nights!

Simply incredible at
£29.75

(Please quote when ordering whether Spectrum or ZX81 owner)

16K RAM Pack for ZX-81	£19.75
64K RAM Pack for ZX-81	£44.75

Prices include VAT, postage & packing. Delivery normally 14 days. Export orders at no extra cost.

Dealer enquiries welcome

Cheetah, products available from branches of

John Menzies  WHSMITH  Rumbelows
and all good computer stores

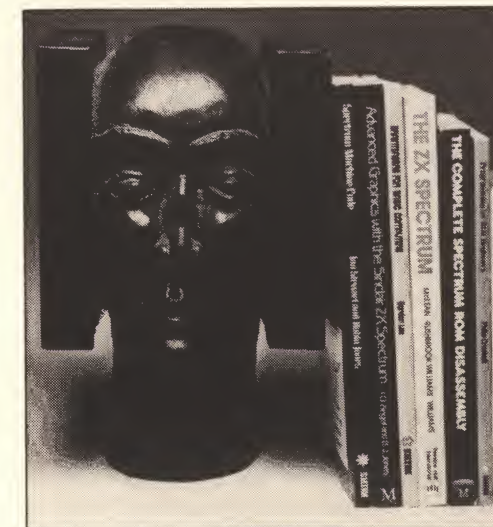
Cheetah
Marketing



Dept AN
Cheetah Marketing Ltd, 24 Ray Street,
London EC1R 3DJ. Tel: 01 833 4909.
Telex: 8954958.

Chapter 9

Books



Each new development in the computer world arrives with, and is often preceded by, an avalanche of hard copy. There are, for instance, more books devoted to the QL than there are QL users to read them. John Gilbert, occasionally guilty of contributing to this abundance of print, digs deep into the 1984 book mountain and discovers some literary gems.

THE YEAR 1984 shaped the computer book market and gave some of the younger companies, such as Melbourne House and Interface, a maturity which is seldom seen outside the world of fiction publishing. That was due to a maturing of the market. Prices went down, the quality was higher than ever before and writers gave away more secrets about computers than ever before.

Machine code

At the start of the year few would have dreamed of books on machine code which would detail the whole inside story of the Spectrum and not just waffle about what the instructions do and little else.

Interface was first with a highly-acclaimed series of books on machine code, one of which is *Mastering Machine Code on Your ZX Spectrum* by Toni Baker. Former owners of ZX-81s may notice that the book is an update on the ZX-81 machine code book. The contents are similar to the earlier version, although the Spectrum book is longer because of the introduction of extended chapters on printing characters on the screen and an update on advanced features on the Spectrum. There the differences end.

The examples in the text, which usually consist of whole chapters, are the same as in the ZX-81 version. They include a graffiti program which displays the Spectrum character set and three chapters which show how a game of draughts is put together.

One useful part is a long chapter which provides a dictionary of all the instructions you can give to the Z-80 processor. It turns the book into a

useful reference guide, which should be of use to all beginners. It will, however, be of little use to advanced programmers — unless they want to brush-up on their terms.

The book also contains some useful appendices which give tables of conversions between hexadecimal and decimal and also a list of symptoms variables which can be used either in machine code or in Basic programs.

Two other books from Interface form a series. *Spectrum Machine Code Made Easy, Volume One* is for the beginner. It is easy to follow and starts, as with most other machine code books, with an introduction to binary and hexadecimal number systems.

The layout of the beginners' book seems to have been thought through carefully, as each chapter deals with one of the major aspects of machine code and, unlike some other books on the market, does not disconcert the reader by moving into another realm halfway through a chapter.

Most of the aspects of machine code covered in Baker's book are included but this is better as it is easier to read and the layout of text and illustrative examples is better.

The companion volume is *Spectrum Machine Code Made Easy, Volume Two*. It is for the advanced programmer and takes it for granted that you have some knowledge of machine code. Because of the complexity of jumps, both relative and absolute, in programs and the specialised logical operations, there are separate chapters on those subjects.

That is as far as the book goes along conventional lines, however, as the author, Paul Holmes, explains the use of ports and interrupt modes. They are

two subjects which are rarely mentioned in books of this kind.

The ports are dealt with in Basic and machine code and some interesting effects are obtained with sound and colour on the Spectrum.

No previous pure machine code book has dealt with interrupts in the Z-80 system. This one explains the subject easily and competently. The chapter on interrupts justifies what is anyway a good book.

Not to be outdone, and rushing in where angels fear to tread, Usbourne Books decided to produce a book on machine code which was aimed not only at the complete beginner but one who is also within the early teenage range. It is *Machine Code for Beginners*. It is the easiest book on the subject on the market. Using it, a child at junior school could learn the basic techniques of both Z-80 and 6502 programming. The way the subject is tackled is nothing short of brilliant. That both major types of microchip are covered is an extra bonus.

The cartoon characters, such as robots and bugs, which are included, teach machine code in a subtle manner. It is possible to read the book as a comic strip, taking little account of the main text and still derive a knowledge of assembly and machine language.

New technology

The same company did a remarkable job of introducing children to the concepts of information technology and the world of robots. *The Information Revolution* contains information about all kinds of devices, including computers, which can be used to communicate, from one part of this planet to another, and even into outer space. It looks into

the future and examines the introduction of cable communication devices within cities, so that you could see people as well as talk to them on the telephone and do your shopping at home using a computer. Much of what is discussed in this colourful book may seem like science fiction but most of it is available now.

Robotics

A second book from Usbourne is *Practical Things To Do With A Microcomputer*. It investigates robots, shows how computers create cryptograms or codes, and illustrates how to write programs using almost any computer you might possess.

It will introduce computer technology to a child and, if the child already knows about computers, it should expand horizons even more. Some of the examples may be a little difficult to understand but, with help from parents, children should cope.

Languages

New computer languages became available on the Spectrum in 1984. Sinclair Research brought out Forth, Micro Prolog and LSCI Logo. The book publishing industry stepped in to give the user a wider base of ideas from which to launch into the brave new worlds of those languages.

Melbourne House was one of the first Sinclair orientated companies to bring out a book about Forth. The company had its own reasons for publishing it, however. *Advanced Spectrum Forth* was launched to help people who bought the company's Abersoft Forth package and did not know what to do with it.

The book, by Don Thomasson,

shows the advanced aspects of the language but it is not only for the complete expert, as the title suggests. It offers explanations of the machine stacks and Forth words which the manual does not begin to cover.

The first half is a re-run of the manual but is written in terms which mere mortals can understand. It also shows the difference between Spectrum Forth and the FIG standard. Those additions to the language are mostly for graphics and sound which, the author says, can be manipulated a great deal more easily in Forth than in Basic.

The second half of the book provides the real interest for people at advanced level. It shows how to structure programs using the screen editor and explains the difficult problem of SAVEing screens to tape.

The book also includes the use of assembly language and how to incorporate it in the structure of Forth programs.

Information about the Logo language was published by Granada in mid-year in a book called *Introducing Logo* by Boris Allen. Apart from *Mind Storms*, written by the creator of Logo Seymour Papert, it is one of the best books on the subject available. The first part of the book provides a clear, concise, introduction to Logo theory together with examples of Logo programming in practice.

The book is ideal for use by teachers who are about to teach Logo or who want some ideas as to how to develop the language. One of the best aspects of it is that Allen does not stop at the simple commands. He goes on to expand the subject and show how Logo can be put to good use by anyone who wants to learn how to program or how

to use mathematics in an adventurous way.

Fiction

Fiction has also found its way into the microcomputer publishing market. Fontana Armada released the first in a series of highly original who-dunnit books. *Input an Investigation*, by Lois and Floyd McCoy, provides a series of classic mysteries featuring those American computer whizz-kids, the Bytes Brothers.

Those brain boxes work their way through five mysteries and always get the correct solution using their home computer called Nibble.

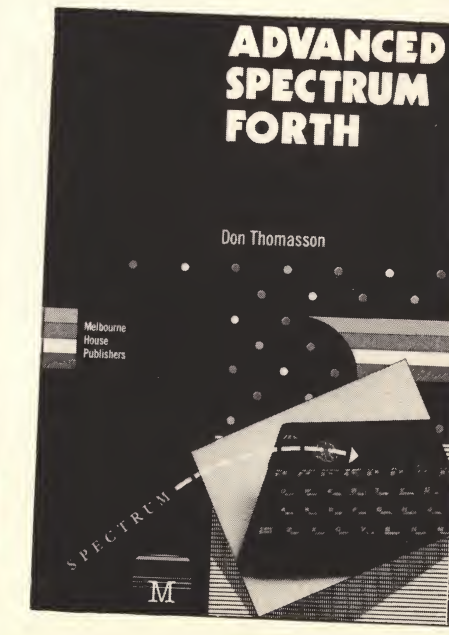
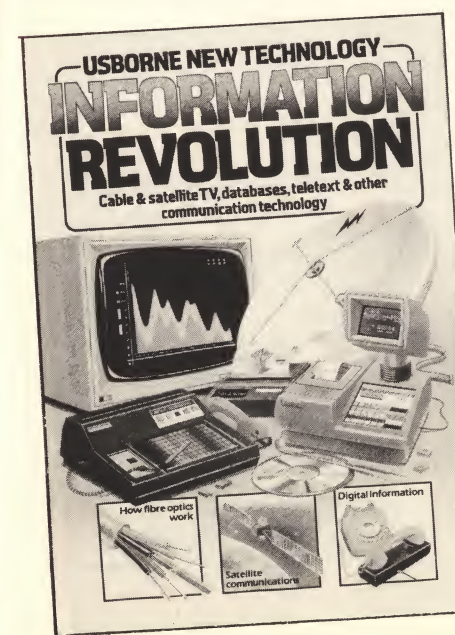
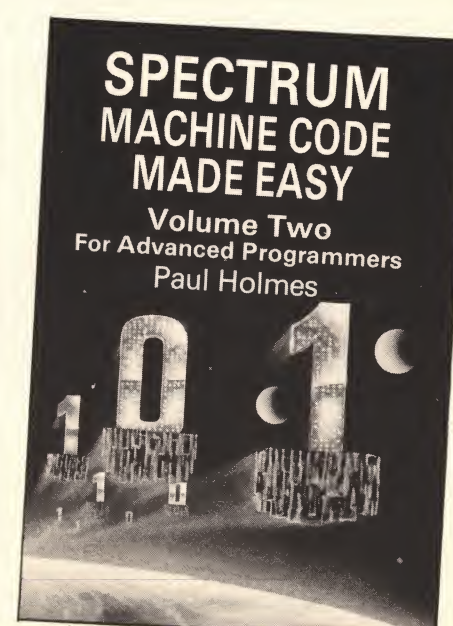
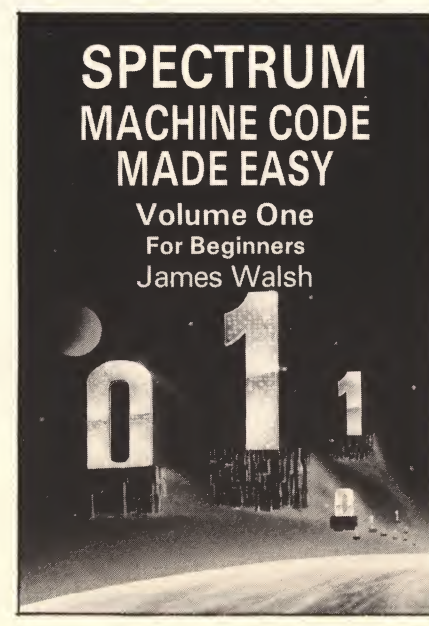
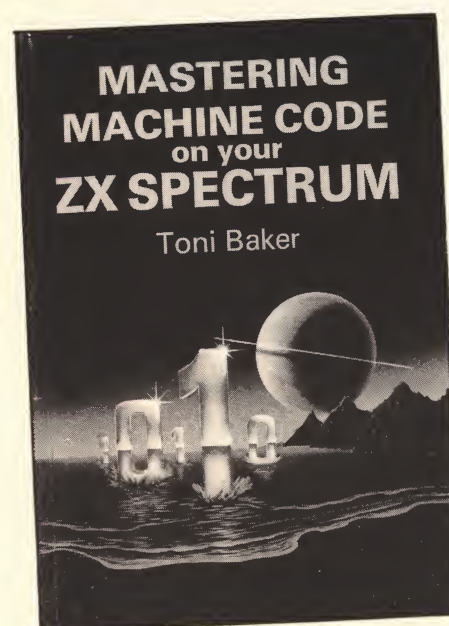
As with any good detective story all the clues for the solution have been incorporated in the plot. The computer programs used to solve the crimes are also included, along with a full explanation on their structure and creation.

The book is for children but adults will be equally enthralled. It provides excellent entertainment and value for money.

General interest

The market for general computer use was just as alive as that of specific usage. One of the best on the market, though it was not perfect, was produced by Penguin. It was one of the few computer book successes which they had in 1984.

The Penguin Computing Book, by Susan Curran and Ray Curnow contains a step-by-step guide to computers and electronics. The criticism is that there is nothing new in the format, although the authors have tried to provide a simple and concise explanation of almost everything relating to digital



Books

electronics, analogue electronics and computers.

Reading the book it appears that the authors have tried to cram too much information into a small space. Any difficulty in following arguments will not be because of any complexity in the text but because so much information is covered in a few pages.

The structure of the book suggests that it should be read from cover to cover, like *War and Peace*, but it serves better as a handbook to be dipped into when necessary. The index is just adequate for such a purpose. The glossary and bibliography are also useful.

QL

The QL, of course, received its fair share of advance publicity from publishers who stumbled onto the market with books as soon as they could.

The first of those titles was the *QL Companion* by Boris Allen with which some reviewers had a field day. Much of its content was irrelevant and many of its SuperBASIC programs did not work. Indeed some bore no relation to SuperBASIC at all.

Thankfully we are seeing books of a better standard from companies such as Sunshine and Longman which should fill the thirst for knowledge which most QL users have. It will be a long time before the tide of QL books stops but, as with any new microcomputer, every piece of information about it is bound to be well received.

Humour

Not all the new books on the market took such a serious look at computers. Two from the States took the opposite viewpoint.

Micromania, by Charles Platt and David Langford, takes computing to pieces and finds the comedy under the skins of big business and the individual computer user. It does so by using a hatchet of slapstick humour with an ungainly mixture of clichés and buzzwords.

The book provides a compendium of insights into the personalities of those who use computers. The authors may be interested in the machines but they are ready to make fun of the behaviour of their fellow maniacs.

They divide users into four classes — the hardware freak, the video game nut, the end-user and the hacker. The descriptions of those people are as cruel as the names used to label them. A video games junkie is, for example, described as 'between 10 and 16 years of age, mentally if not physically'.

The structure of the book is inventive, to say the least. Each chapter is labelled with a binary number, beginning with 0000 and rising to 1101. The text is split into modules reminiscent of the way in which programs should be structured. It also shows that the authors are part of the computer world of which they are making fun.

Chapter 1101 deals with the future of micromputing, a subject which would have been a glaring omission from a book dealing with the whole computer scene. The authors may ridicule the denizens of the computer industry but they agree that our future is in their hands:

"We can laugh all we like at micro-maniacs, but they will have the last laugh — because they are designing the future that the rest of us will have to live in."

The Naked Computer, by Jack Rochester and John Gantz starts by looking at the computer invasion and the most successful and unsuccessful computer ventures.

The most ubiquitous computer? The Commodore Vic-20, one million strong at the end of 1982. The least ubiquitous? The CDC G-20, of which there is one left.

The final chapter, *The Outer Limits*, is like *Micromania* in that it tries to explain what could happen in the next few decades so far as technology is concerned. The book looks at new IBM projects, research into chip use done at Massachusetts Institute of Technology and the way in which robots are being used by the US Army.

Like all the other chapters, *The Outer Limits* is a collection of anecdotes, although some of the authors' thoughts are included. The book leaves the reader with the feeling that the authors were afraid to approach the subject in anything but the impersonal third person.

Mastering Machine Code On Your ZX Spectrum, Spectrum Machine Code Made Easy Volumes One and Two. All titles £5.95. Interface Publications.

Machine Code for Beginners, The Information Revolution, Practical Things To Do With A Microcomputer. All titles £1.99. Usbourne Books.

Advanced Spectrum Forth, £8.95. Melbourne House.

Introducing Logo. Granada.
Input an Investigation. £1.25. Fontana Armada.

The Penguin Computing Book, £5.95. Penguin Books.

The QL Companion, £6.95. Pitman.
Micromania, £7.95. Victor Gollancz Publications.

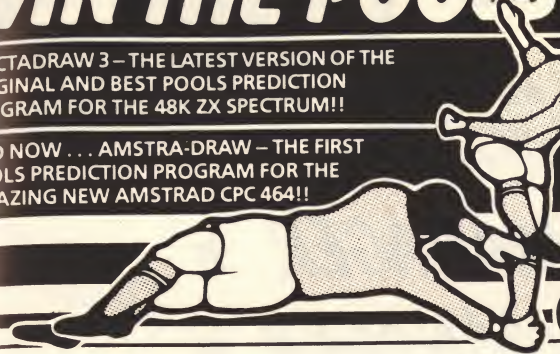
The Naked Computer, £9.95. Arlington Books.

★ NOW AVAILABLE FOR THE AMSTRAD CPC 464!!

WIN THE POOLS?

SPECTADRAW 3 – THE LATEST VERSION OF THE ORIGINAL AND BEST POOLS PREDICTION PROGRAM FOR THE 48K ZX SPECTRUM!!

AND NOW . . . AMSTRA-DRAW – THE FIRST POOLS PREDICTION PROGRAM FOR THE AMAZING NEW AMSTRAD CPC 464!!

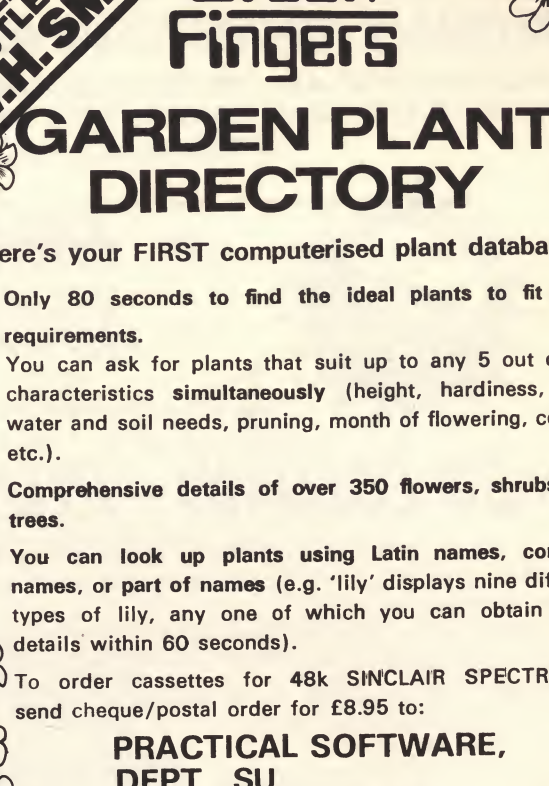


- Supplied with Database containing data on over 10,000 matches since 1980!
- You update the Database each week – but no tedious typing, as team and division names already in program!
- Errors easily corrected – the program even checks your entries!
- Comprehensive instruction manual and menu driven program – easy to use, even for a newcomer to computing!
- Will forecast the least likely draws for those who prefer to bet on fixed odds!
- Built in perm generator – complete your coupon direct from the screen!
- Fully microdrive compatible! (Spectadraw only).
- Compatible with Currah Microspeech – the first pools program to read you its predictions! (Spectadraw only).

Spectadraw 3 for the 48K Spectrum.....£9.95 inclusive
 Amstra-Draw for the Amstrad CPC 464£9.95 inclusive
 (Cheques/P.O.s payable to B. S. McALLEY)

We dispatch every Monday with the database made up to include all matches up to the date of dispatch.

SPECTADRAW (Dept SUA), 1 Cowleaze, Chinnor, Oxford OX94TD.
(Tel: 0844-52426)



**NOW IN
SELECTED
OUTLETS OF
W.H. SMITH**

Green Fingers

GARDEN PLANT DIRECTORY

Here's your FIRST computerised plant database!

- * Only 80 seconds to find the ideal plants to fit your requirements.**
You can ask for plants that suit up to any 5 out of 17 characteristics **simultaneously** (height, hardiness, sun, water and soil needs, pruning, month of flowering, colour, etc.).
- * Comprehensive details of over 350 flowers, shrubs and trees.**
- * You can look up plants using Latin names, common names, or part of names** (e.g. 'lily' displays nine different types of lily, any one of which you can obtain fuller details within 60 seconds).

To order cassettes for 48k SINCLAIR SPECTRUM send cheque/postal order for £8.95 to:

**PRACTICAL SOFTWARE,
DEPT. SU,
40 WORPLE ROAD,
LONDON, SW19.**

THE KET TRILOGY

'A commendably strong plot and a pot full of puzzles'
—BIG K

'Full marks'
—CEVG

'100% — the best I have ever reviewed'
—PCT

'A stimulating adventure'
—SINCLAIR USER

'Heartily recommended'
—CRASH

'Very professional ... a very polished adventure'
—PCW

A VIDEO RECORDER

WHEN YOU BECOME BRITAINS' 'BEST ADVENTURER'

KET is a strife torn land which has never known peace. Particularly vicious attacks from beyond the mountains now threaten its very existence and the Lords of Ket look upon you as their only hope ...

Each episode of the Ket Trilogy hides a short part of a sentence that is only revealed on completing the adventure. Having come to the end of this mammoth 120K challenge, the first person to discover the complete message will be awarded a video recorder of their own choice (up to value of £400) and the coveted award BRITAINS BEST ADVENTURER.

The Complete Ket Trilogy now Available in a Presentation Box. Special Price £12.95

THE KET TRILOGY Series of adventures are available from Software Dealers nationwide. In case of difficulty, please use our fast and efficient mail order service.
PLEASE NOTE Each of the Trilogy is a **COMPLETE ADVENTURE IN ITSELF** which can be played totally independently of the other two.
REGISTERED OFFICE 54 London Street Reading RG1 4SQ.
CREDIT CARD ORDERS Telephone direct (0734) 591678.

ORDER FORM

ZX SPECTRUM 48K
£5.50 each (incl. P&P)

Mountains of Ket
Temple of Vran
The Final Mission

Also available:
1984 — Government ☐
Management ☐
Millionaire ☐ Splat ☐

THE KET TRILOGY PRESENTATION BOX £12.95

Please send me the titles as indicated, by 1st class post.

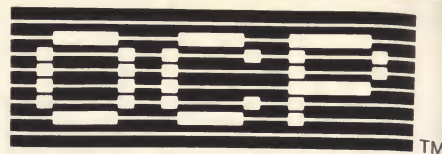
I enclose cheque for £ or debit my credit card.

Access ☐ Visa ☐
Barclaycard ☐

Name/address

INCENTIVE SOFTWARE LTD, 54 London Street, Reading RG1 4SQ, England.

48K SPECTRUM OWNERS ESPECIALLY MICRODRIVE OWNERS



OCP ANNOUNCE ALL PROGRAMS COMPLETELY REWRITTEN FOR ZX MICRODRIVE
COMPATIBILITY & PLUS 80's FOR USE WITH 15 DIFFERENT CENTRONICS/RS232
INTERFACES INCLUDING ZX INTERFACE 1



★ FINANCE MANAGER ★ ★ ADDRESS MANAGER ★

£8.95

(BOTH LESS £3 REBATE FOR EXISTING USERS ON RETURN OF OLD CASSETTE)

★ ★ 80 COLUMN-PLUS 80 VERSIONS ★ ★ £19.95

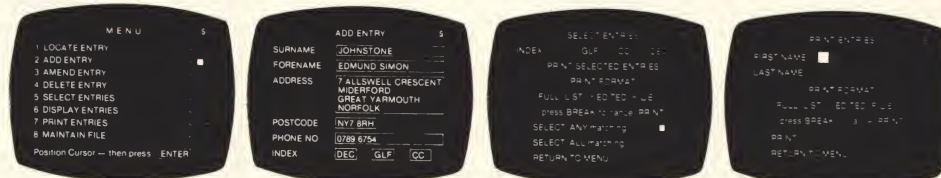
(LESS £5.00 REBATE FOR EXISTING USERS ON RETURN OF OLD CASSETTE)

Write to us for a quotation for the software, the interface and the latest high performance Japanese Dot Matrix Printer — you will find our prices very competitive.

Both programs are available in standard form which work only with ZXTM and Alphacom 32 column printers, and Plus 80 which work in conjunction with most Centronics/RS232 Interfaces and 80 column printers.

ADDRESS MANAGER and **FINANCE MANAGER** utilise the same "on the page" presentation and offer 48K Spectrum owners a professional standard address filing, indexing retrieval, and financial analysis system. Below are examples of the screen presentations.

Both have been carefully constructed to provide the user with a tool that is extremely friendly and easy to use, the speed and presentation of these programs are second to none.



ADDRESS MANAGER features MULTIPLE INDEXING via our 3 way 3 character index, an ability to store over 400 full names and addresses or 1500 individual names/titles.

USES include storing and updating names, addresses and phone numbers, printing out Xmas card lists, etc, mail order work, customer classification by type size (doctors have used this program to catalogue patients by treatment).

FINANCE MANAGER is a powerful, flexible and fast MENU DRIVEN general purpose program carefully designed to handle up to 255 separate accounts for domestic and business accounting applications. The magic of MACHINE CODE has enabled us to produce the very latest "on the page" presentation which lets you enter and edit data naturally, as if with a pencil and paper.

Voted "The best value financial program available" by Sinclair user.



But that's not all, not by a long way. This program automatically raises a corresponding debit or credit for every entry, and will even open up a new account if an entry features an unrecorded account name.

Accounts can be MERGED, DELETED, ANALYSED, MARKED as priority, RENAMED, EDITED and SCROLLED. Transactions can be RECONCILED, AMENDED, DELETED, PRINTED, DESCRIBED for analysis and RENAMED. Standing orders can be APPLIED, REMOVED, DESCRIBED, AMENDED, DELETED and even DUMMIED for planning purposes. Other features include DATE CHANGE, RUNNING TOTALS, 2 KEYBOARD MODES, PRINT PAGE/ LINE/BLOCK/FROM END/FROM START/FROM DATE etc., LIST BALANCES, FIELD ERASE/INSERT/DELETE, EXIT TO BASIC. You may not want all these features but they are there just in case.

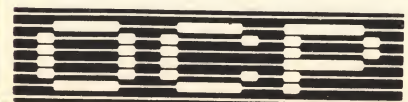
★ ★ ★ ★ UTILITIES ★ ★ ★ ★ £9.95

- ★ MACHINE CODE TEST TOOL ★
- ★ EDITOR ASSEMBLER ★
- ★ MASTER TOOL KIT ★

Sinclair ZX MICRODRIVE COMPATIBLE

(Existing users wanting latest versions will receive a £3.00 rebate on return of old cassette)

Until stocks reach WHSMITH and other retail outlets send cheque or telephone details to 0753 889055. Replacement tapes only available from OCP direct.
OXFORD COMPUTER PUBLISHING LTD., 4 High Street Chalfont St. Peter, Bucks. SL9 9QB.



SOFTWARE ★ ★ ★ ★ SIMPLY THE BEST

SU/12/84

Chapter 10

Starter Pack



With only a minimum of programming expertise it is possible to enter the listings published in *Sinclair User* every month.

A study of the way in which such programs are constructed will, however, be invaluable when the reader comes to write his own.

With that end in view we reprint a selection of line-by-line program explanations which appeared monthly in the Starter Pack section of *Sinclair User*.

Variables used by the program:

- S = score.
- A = landing pad position across screen.
- B = ship's position across screen.
- C = ship's position down screen.
- Z = landing pad position down screen.
- 10 Make the starting score 0 and . .
- 20 Clear the screen.
- 30 Print the score.
- 40 Start the landing pad at the left of the screen.
- 45 Find a random position for the landing pad down the screen which is at least seven lines down but not more than 20.

50 Start the ship in the middle of the screen.

This is the start of the main play loop. The program works its way through all the lines to line 140 and then returns to that point and loops through again, unless told to do something else.

60 Increase the position of the landing pad across the screen.

70 Find the new position of the ship across the screen. If you are pressing the 8 key, the expression (INKEY\$="8") will be true and will equal 1; if not, it will equal 0. The same applies to the 5 key. If both are being pressed, the INKEY\$ function will not work, as it can recognise only one key at a time. That kind of mathematics is known as Boolean algebra; for more information see chapter 10 of the manual.

80 Increase the position of the ship down the screen.

90 Prints the trail left by the ship.

100 Prints the ship.

110 If the across and down positions match, the ship has landed, so go to the extra lines which increase the score and start a new game.

120 If the ship has reached the bottom of the screen, go to the end of the game.

130 Prints the landing pad.

140 Go back to the start of the play loop.

The increase score routine. If you landed in line 110 the program goes to line 150 to increase the score and start a new game. When writing a program it is best to put all routines which are not often used outside of the main play loop. In that way the game will be a little faster, as it does not have to look through so many lines.

150 Increase the score by 10.

160 Returns to the start of the game but does not re-set the score.

170 The end of the game. Print the crashed ship.

AS YOUR starship hurtles to the ground leaving behind a trail of asterisks, you must attempt to dock on the moving landing pad. Use key 5 to move left and 8 to move right. **Lander** was written for the 1K ZX-81 by Colin Flisk of Altrincham, Cheshire.

```

10 LET S=0
20 CLS
25 PRINT AT 0,0;S
30 LET C=0
40 LET A=1
45 LET Z=INT (RND*13)+7
50 LET B=10
60 LET A=A+1
70 LET B=B+(INKEY$="6")-(INKEY$="5")
80 LET C=C+1
90 PRINT AT C-1,B-1;" * "
100 PRINT AT C,B;"███"
110 IF B=A AND C=Z-1 THEN GOTO 150
120 IF C=20 THEN GOTO 200
130 PRINT AT Z,A;"███"
140 GOTO 60
150 LET S=S+10
170 PAUSE 50
180 GOTO 20
200 PRINT AT C,B;"███"

```

LANDER
LANDER
LANDER
LANDER
LANDER

PANIC by Aaron Day squeezes a fast-moving arcade game into the 1K ZX-81. You must steer your helicopter through the obstacles and land it on the mountain, using cursor keys 6 and 7 to control your machine. Day has used a number of sophisticated space-saving techniques to fit the program to the memory, including a short machine code routine. You must enter that separately.

Type in line 1 of the program and then type the following: POKE 16514,42 NEWLINE POKE 16515,14 NEWLINE POKE 16516,64 NEWLINE POKE 16517,78 NEWLINE POKE 16518,6 NEWLINE POKE 16519,0 NEWLINE POKE 16520,201 NEWLINE. Then type in the rest of the program. You will see that the REM statement contains a jumble of characters; that is the little machine code routine.

```

1 REM XXXXXXXXX
5 FAST
10 FOR F=SIN PI TO CODE "G"
20 PRINT AT RND*10,RND*22+2;"███"

30 NEXT F
40 PRINT AT 10,25;"███";AT
9,26;"███";AT 8,27;"███"
45 SLOW
50 LET X=CODE "."
60 LET Y=SIN PI
70 PRINT AT X,Y+1;
80 LET P=USR 16514
90 PRINT AT X,Y;"███"
100 IF P=136 THEN PRINT "CRASH"
110 IF P=136 THEN STOP
120 PRINT AT X,Y;"███"
130 LET Y=Y+1
140 IF Y<CODE "." THEN GOTO COD
F "COS "
150 IF X=CODE "F" THEN GOTO COD
F "█"
160 PRINT AT NOT PI,NOT PI;"YOU
MISSED THE MOUNTAIN"
170 STOP
180 PRINT AT 2,NOT PI;"YOU LAND
ED"
190 STOP
200 LET X=X+(INKEY$="6")-(INKEY$="7")
210 GOTO 70

```

In Sinclair Basic it takes much more memory to hold a number than a letter — six bytes instead of one. Day avoids that by using expressions such as CODE "G" or NOT PI which are numerical values. In the appendix to the ZX-81 manual you can find a full list of all the CODE values. See if you can determine what numbers the other expressions represent.

- Line 1. The REM statement contains the machine code routine. Last month we published a program — Frogger — which used a Basic line to read the screen and see if there was anything on-screen at the point where the next character was to be printed. The machine code routine here does the same thing in even less space.
- Lines 10-30. Print-out the obstacles at random over the playing area. The character in line 20 is a grey square. You can write it by typing SHIFT H in graphics mode.
- Line 40. Prints the mountain.
- Lines 50-60. Set the initial co-ordinates of the helicopter.

- Line 70. Gets ready to print the helicopter.
- Line 80. Calls-up the machine code routine in the REM statement, which looks at the space to print the helicopter. If that space has an obstacle, P will be given the value 136.
- Line 90. Prints the helicopter.
- Lines 100-110. Check for a crash. If P=136 you have hit an obstacle.
- Lines 120-130. Print a space where the helicopter was and increase the Y co-ordinate. That is necessary because the helicopter is two characters long.
- Lines 140-190. Check your position relative to the mountain. The mountain is always in the same place, so there is no need to use the machine code routine. If you have landed at the proper point or missed it completely the game stops.
- Line 200. The INKEY\$="6" expression has the value 1 if you are pressing key 6, similarly for INKEY\$="7". Otherwise it is zero. The whole line adjusts the co-ordinates by one depending on which key you are pressing.
- Line 210. Returns to line 70 to re-start the game process.



SLEEPING BEAUTY

```

2 LET B=NOT PI
3 LET S=NOT PI
5 LET X=INT (RND*CODE "A")
6 IF X>CODE "2" THEN LET X=-X
7 LET Y=INT (RND*CODE "A")
8 IF Y<CODE "2" THEN LET Y=-Y
9 LET D=SQR (X*X+Y*Y)
10 PRINT INT D;"km->castle"
11 PRINT "N/S/E/W?"
12 GOSUB CODE "A"
13 LET X=X+(D$="N")-(D$="S")
14 LET Y=Y+(D$="E")-(D$="W")
15 IF X=NOT PI AND Y=NOT PI TH
EN GOTO CODE "9"
16 LET D=INT (RND*CODE "$")+PI
/PI
17 LET D$=("OGRE" AND (D<VAL "
6"))+("SWORD" AND (D>VAL "5")) AN
D (D<VAL "9"))
18 IF D>=VAL "9" THEN GOTO VAL
"9"
19 PRINT "found";D$;"n\1"
20 IF D$="OGRE" AND S=PI/PI TH
EN GOTO CODE "0"
21 IF D$="SWORD" THEN LET S=PI
/PI
22 IF D$="SWORD" THEN GOTO VAL
"9"
23 IF D$="OGRE" THEN PRINT "wa
it(7)run(6)"
24 GOSUB CODE "A"
25 IF D$="7" OR D$="6" THEN GO
TO CODE D$
26 IF D$<>" " THEN GOTO CODE "7
"
27 GOTO VAL "9"
28 LET S=NOT PI
30 LET B=B+PI/PI
31 PRINT "OGRE DEAD BY SWORD"
32 GOSUB CODE "A"
33 GOTO VAL "9"
34 IF RND*VAL "9">PI/PI THEN G
OTO VAL "9"
35 PRINT "dead"
36 STOP
37 PRINT "princess
wake";B;"d
ead ogre"
38 INPUT D$
39 CLS
40 RETURN

```

SLEEPING BEAUTY is a 1K adventure game by Kevin Wright of Northgate, Hull who wrote it because his friends believed a 1K adventure game written in Basic was impossible to program. The object is to reach the princess who is sleeping in the castle. You must kill the ogres who are placed along your route by the wicked fairy. You are armed with the swords which are distributed by good fairy. Lower-case characters represent inverse letters.

The variables

A VARIABLE is a location within the computer memory which stores a value such as a number or a string of characters.

The variables used in the program are:

B the number of battles fought and won.
S if you own a sword then 'S' will equal 1. Otherwise, S will equal 0.
X and Y Those contain your co-ordinates on the castle floorplan grid.
D That will be used for different purposes during the program, but it is used mostly to hold the value of your distance from the castle.
D\$ A chain or string of characters which indicates the direction in which you are travelling.

The Program lines

2 and 3 make the values of B and S equal to 0.
5-8 put random position numbers in X and Y.
9 uses the square root function to find your distance from the castle.
10 and 11 display your position and the directions in which you can move on the screen.
12 calls the subroutine, a program within a program, starting at line 38. That routine will prompt you to enter a string of characters which will be put into the variable D\$.
13 and 14 find your new position on the map grid.
15 if X and Y both equal 0 then you have reached the castle. If that condition is true, go to line 37; if not, continue to next line.
16 finds a random integer between 1 and 13.
17 if D equals a number between 1 and 5 put "OGRE" into D\$. If D is between 6 and 8 put "SWORD" into D\$.
18 if D is greater than 9 then go to line 9.
19 print what you have found on the screen.
20 if you encounter an OGRE and you have a SWORD then go to line 28.
21 and 22 if you found SWORD put 0 into S and then go to line 9.
23 and 24 if you met OGRE and you have a SWORD then the options available are displayed and prompt for a response.
25 go to line 34 or 35 depending on the number you get from the input, entry, routine.
26 if you have not put in a response allowed by the computer go to line 35.
28-30 make S equal 0 and increase B by one for the number of battles won.
31 and 32 display a message and wait for NEW LINE.
34 and 35 depends on lines 25 and 26 and gives you a random chance of dying.
36 STOPS you as you are dead.
37 congratulations and number of battles won.
38-40 player-response entry routine used in line 12.

frogger

Our second program from Neil Palmer of Barrow-in-Furness is a 1K version of **Frogger** for the ZX-81. You must guide the frog across the road without being run over by the juggernauts. Use keys 5 and 8 to move left and right, and key 7 to go forward.

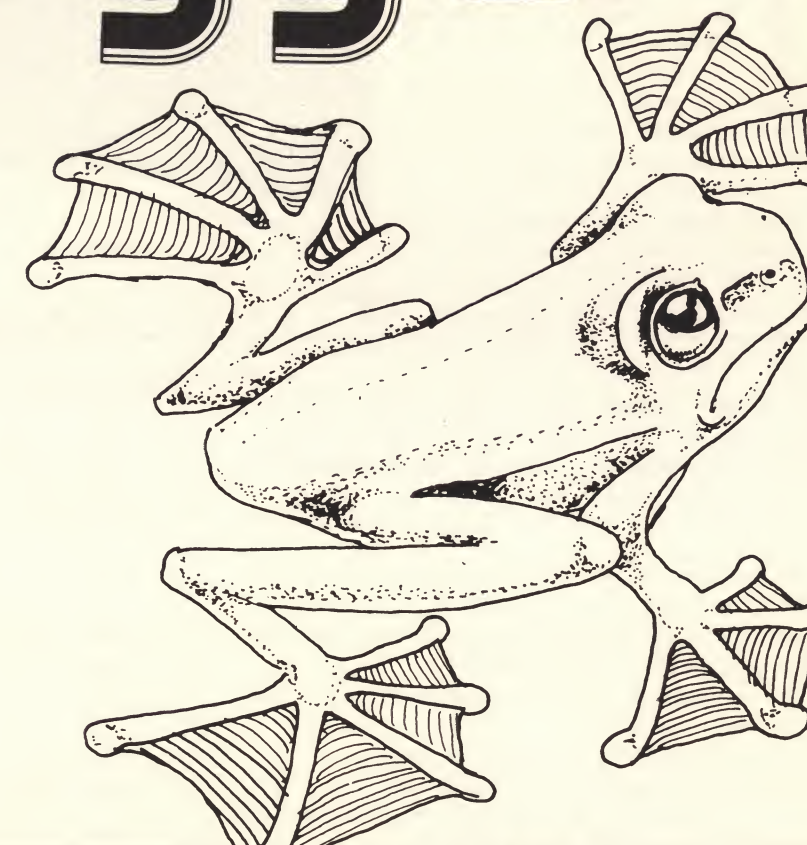
Variables

A variable is a location in memory where information can be stored. Usually the information will change as the program runs. To help you understand Frogger, here is a list of the variables used:

A\$,B\$: Hold the picture of the lorries.
A, B Hold the co-ordinates for the position of the frog.
C\$ Holds the number of the key you pressed to move.
S Holds your score.

Program

10-60 Set up the picture on the screen and your starting position. INKEY\$ is a command to read the keyboard and see if a key is being pressed.
70 Prints the picture on the screen. AT A, B; gets ready to print the frog at the proper place.
80 This complicated line reads information straight from the screen. It looks — PEEK — in the stores the computer sets aside for the screen and sees what is in there. In this case it is looking at the next place to PRINT AT, which will be the position of the frog. If the value of the character there is more than 127 — in this game it would then be part of a lorry — you have been run over and the program will STOP. You can find these values, called codes, in the back of the ZX-81 manual. This kind of PEEK command is a very useful way of checking where you are moving in an arcade game and saves a good deal of memory if you have only 1K.
100 Sends you to a special routine if you want to go forward.
110 Works out your new co-ordinate if you want to go sideways. You could do that with separate IF statements for left and right but Palmer has combined the two cleverly by using logic operations. The computer works out if the expressions C\$="8", B<11 and the like are true and then if the whole expression with AND is true. If it is, the value of the bracketed expression is set to 1 and added or subtracted from the co-ordinate for right and left.
120 Prints the frog at the new position.
130-150 Work out the new picture by chopping off the front column and putting it on the other end; the program then returns to line 60 to start the routine again.
160-220 Let you move forward. Every time you go forward S is increased for your score and printed. If you win — 190-220 — the computer re-sets the starting position and begins again.



```

10 LET A$=" "
20 LET B$=" "
30 LET S=0
40 LET A=9
50 LET B=6
60 LET C$=INKEY$
70 PRINT AT 1,0;A$,,,B$,,,A$
,,,B$;AT A,B;
80 IF PEEK (PEEK 16398+256*PEE
K 16399)>127 THEN STOP
90 PRINT AT A,B;" "
100 IF C$="7" THEN GOSUB 160
110 LET B=B+(C$="8" AND B<11)-(
C$="5" AND B>0)
120 PRINT AT A,B;"F"
130 LET A$=A$(2 TO )+A$(1)
140 LET B$=B$(12)+B$( TO 11)
150 GOTO 60
160 LET A=A-1
170 LET S=S+(5)+(100 AND A=0)
180 PRINT AT 12,0;S
190 IF A>0 THEN RETURN
200 LET A=9
210 LET B=6
220 RETURN

```


COSMIC MISSION

T. SIMMONS of Downham Market, Norfolk wrote **Cosmic Mission** for the 1K ZX-81. Guide your craft through outer space using keys 5 and 8. If you bump into the enemy your force field will disappear and you will be spotted. The score will be given and you can then run the program again. The explanation of the program lines and an adaption for the Spectrum is given below.



100 Defines the Enemy
105 Sets the score to zero for start of game
110-125 These variables are used later in the game and are set to zero initially (see lines 215-250)
130 The amount you score is increased each go. Initially it is set to 1
135 The starting position of your craft across the screen
160-300 This series of lines controls the movements in the game. The program will loop around these lines until the conditions in line 270 have been met.
160 Chooses a random position across the screen for the enemy between 0 and 26
170 Prints the enemy at the bottom of the screen
180-190 Scrolls the whole screen up by one line
200 Each time the program passes this point in the loop your score is increased by the value of t
210 If you have scored 104 this increases the value of t
215-250 r is the position across the screen of the enemy. On the first all these variables are at zero, and as the program passes line 250 a takes the value of r. On the second time round, as the enemy scrolls up the screen, b has the first value of r and a now has the second, and so on until the fifth time when e has the first value of r. By this time the enemy has scrolled up to your position. See line 270
255 Blanks out your craft which has scrolled up two lines from its previous position
260 Redraws your craft
270 Checks to see if your position x is within three squares — four if t=2 — of the enemy, and if it is a jump is made to line 500
280-290 These move your craft left or right to the value of t depending on which key you are pressing
300 Jumps back to the start of the loop
500 Prints that you have been spotted at your current position
510 Prints your score
515 Invites you to try again

To make this program work on the Spectrum replace lines 180 and 190 with:

```
180 POKE 23692,3; PRINT
190 PRINT
```

If the game is then too fast, add:
295 FOR F=1 TO 50 NEXT F

```
100 LET A$="███"
105 LET N=0
110 LET A=0
115 LET B=0
120 LET C=0
125 LET D=0
130 LET T=1
135 LET X=12
150 LET R=INT (RND*27)
170 PRINT AT 21,R;A$
180 SCROLL
190 SCROLL
200 LET N=N+T
210 IF N=104 THEN LET T=2
215 LET E=0
220 LET D=C
230 LET C=B
240 LET B=A
250 LET A=R
255 PRINT AT 9,X-2;"_
260 PRINT AT 11,X;"███"
270 IF X>=E-2 AND X<=E+T THEN G
OTO 500
280 IF INKEY$="5" THEN LET X=X-
T
290 IF INKEY$="8" THEN LET X=X+
T
300 GOTO 180
500 PRINT AT 11,X-1;"SPOTTED"
510 PRINT AT 0,0;"SCORE=";N
515 PRINT "TRY AGAIN"
```

CIRCLE PATTERNS draws up to sixty different patterns based on the CIRCLE command, which is used here in an unusual way. The patterns generated look very like tessellating floor tiles and could be used to good effect in title screens.

The program was written for the 16K Spectrum by D Trebilcock of Sale in Cheshire.

Program Explanation

Variables:

col	the number of the colour used
rad1	the radius of the inner circle
rad2	the radius of the outer circle
amend1	adjustment to the horizontal print position
amend2	adjustment to the vertical print position
dec	size factor for the pattern
c	control for FOR/NEXT loop
a,b	hold data for print positions

Line 20 Sets ink to black and the rest of the screen to white.
Line 30 Sets the colour for the pattern at random.
Line 40 RESTORE tells the computer to start READING data from a particular DATA statement — in this case at line 250. Without the RESTORE command, when the program was executed for a second time the computer would be unable to find any more data to load.

Line 60-70 Set two different figures for a radius of the circle. Depending on the size of each unit in the pattern, there may be more than one circle in each unit.

Line 80-90 Set two separate figures for adjusting where each part of the pattern is to be printed.

Line 100 Sets a factor for the size of the pattern units; 0, 1 or 2.
Line 110 The main loop of the program starts here. There are 23 units to be drawn.

Line 120 READs the print positions from DATA statements starting at 250. When the first statement is used up the computer moves automatically to the next.

Line 130 If the unit size is 1 or 2 the computer draws a small circle. Normally with CIRCLE the circle is a curved line. Here, because INK and PAPER colours are set the same, the whole character square through which the circle passes is set to one colour. This is a peculiarity of the Spectrum use of colour which for once has been turned to good advantage. Try changing either the INK or PAPER colours and see what happens.

Line 140 If the unit size is 0 or 1 the computer draws a larger circle, some of which may obliterate parts of the earlier circle.

Line 150 INKEY\$ checks to see if you are pressing the keyboard. In this case, if you press "S" the program will STOP.

Line 160 The loop returns to line 110 to draw another unit.

Line 170-180 The computer pauses to display the finished pattern and then goes back to line 30 to set a new colour and draw a new pattern.

Line 250-260 DATA statements which contain the co-ordinates for printing the pattern. The program will READ them two at a time and then adjust them slightly for more variety. See what happens if you alter a few of them.

CIRCLE PATTERNS

```
100 LET dec= INT ( RND *3)
110 FOR c=1 TO 23
120 READ a,b
130 IF dec>0 THEN CIRCLE PAPE
R col; INK col;a+amend1,b+amend2
,rad1
140 IF dec<2 THEN CIRCLE PAPE
R col; INK col;a+amend1,b+amend2
,rad2
150 IF INKEY$ ="q" OR INKEY$
="Q" THEN STOP
160 NEXT c
170 PAUSE 90
180 GO TO 30
250 DATA 83,150,131,86,131,22,2
03,118,107,54,59,118,179,86,83,2
2
260 DATA 107,118,35,86,59,54,22
7,22,35,150,179,150,155,54,35,22
,179,22,131,150,83,86,155,118,22
7,86,203,54,227,150
```

```
20 BORDER 7: PAPER 7: INK 0
30 LET col= INT ( RND *7)
40 RESTORE 250
50 CLS
60 LET rad1= INT ( RND *10+4)
70 LET rad2= INT ( RND *7+16)
80 LET amend1= INT ( RND *9-4)
90 LET amend2= INT ( RND *4)
```


SINCLAIR USER *Annual 1985*

software that's hard to beat...

48K
SPECTRUM

TASWORD TWO

The Word Processor *

"If you have been looking for a word processor, then look no further."

CRASH June 1984

"The number of on-screen prompts, together with the excellent manual, make it ideal - even for an absolute beginner."

PERSONAL COMPUTER WORLD

September 1983

"Without doubt, the best utility I have reviewed for the Spectrum."

HOME COMPUTING WEEKLY

April 1984

Your Spectrum becomes a professional standard word processor with TASWORD TWO. Sixty-four characters per line on the screen is just one of the many features of this versatile program. The cassette also contains TASWORD TWO TUTOR. This teaches you word processing using TASWORD TWO. Whether you have serious applications or simply want to learn about word processing, TASWORD TWO and TASWORD TWO TUTOR make it easy and

enjoyable. TASWORD TWO is readily adapted for the microdrives to give super-fast saving and loading of both program and text.

PRICE £13.90

TASMERGE

The Mail Merger

Transfer data from MASTERFILE to TASWORD TWO! Letters and forms typed on TASWORD can be printed with addresses and data taken from MASTERFILE. The mail merge facility allows, for example, multiple copies of a letter to be printed, each containing a different name and address taken from your MASTERFILE data. To use TASMERGE you must have one or more microdrives as well as TASWORD and MASTERFILE by Campbell Systems. (version 9 or later).

PRICE £10.90

TASPRINT

The Style Writer

A must for dot-matrix printer owners! Print your program output and listings in a choice of five impressive print styles. TASPRINT utilises the graphics capabilities of dot-matrix printers to form, with a double pass of the printhead, output in a range of five fonts varying from the futuristic DATA-RUN to the hand-writing style of PALACE SCRIPT. TASPRINT drives all dot-matrix printers with bit image graphics capabilities and can be used to

print TASWORD TWO text files. TASPRINT gives your output originality and style!

PRICE £9.90

TASWIDE

The Screen Stretcher

With this machine code utility you can write your own Basic programs that will, with normal PRINT statements, print onto the screen in the compact lettering used by TASWORD TWO. With TASWIDE you can double the information shown on the screen!

PRICE £5.50

TASMAN PRINTER INTERFACE

Plug into your Spectrum and drive any printer fitted with the Centronics standard parallel interface. Supplied complete with ribbon cable and driving software. The user changeable interface software makes it easy to send control codes to your printer using the method so successfully pioneered with TASWORD TWO. The cassette also contains fast machine code high resolution full width SCREEN COPY SOFTWARE for Epson, Mannesmann Tally, Seikosha, Shinwa, Star, and Tandy Colour Graphic (in colour!) printers. Compatible with microdrives and ZX Interface 1.

PRICE £39.90

TASMAN SOFTWARE

All prices include VAT and post and packaging. Telephone Access Orders: Leeds (0532) 438301
* Available from larger branches of Boots

Tasman

SOFTWARE

Springfield House, Hyde Terrace, Leeds LS2 9LN. Tel: (0532) 438301

If you do not want to cut this magazine just write your order and post to:

TASMAN SOFTWARE, dept. SU, Springfield House, Hyde Terrace, Leeds LS2 9LN.

I enclose a cheque/P.O. made payable to Tasman Software Ltd. OR charge my ACCESS number

NAME

ADDRESS



COMPUTER ITEM PRICE

_____ £ _____

_____ £ _____

_____ £ _____

Outside Europe add £1 for each item

airmail £ _____ TOTAL £ _____

Send me the FREE Tasman brochure describing your products. tick here: ☐

I would like to know more about your programs for:

ZX Spectrum ☐ MSX ☐ Amstrad CPC 464 ☐

Chapter 11

Program Printout

GRAPHICS INSTRUCTIONS

Instructions for graphics characters are printed in lower-case letters in our listings. They are enclosed by brackets and separated by colons to distinguish them and the brackets and colons should not be entered.

Inverse characters are represented by the letter "i" and graphics characters by "g". Thus an inverse W would be represented by "iw", a graphics W by "gw", and an inverse graphics W by "igw".

Spaces are represented by "sp" and inverse spaces by "isp". Whenever any character is to be used more than once, the number of times it is to be used is shown before it, together with a multiplication sign. Thus "6*isp" means six inverse spaces and "(g4:4*i4:g3)" would be entered as a graphic four, followed by an inverse four repeated four times, followed by a graphics three.

Where whole words are to be written in inverse letters they appear in the listings as lower-case letters. Letters to be entered in graphics mode on the Spectrum are underlined.

Inverse characters may be entered on the ZX-81 by changing to graphics mode and then typing the appropriate characters and on the Spectrum by changing to inverse video and typing the appropriate letters. Graphics characters may be entered on the ZX-81 by changing to graphics mode and then pressing symbol shift while the appropriate characters are entered. On the Spectrum graphics characters may be obtained by changing to graphics mode and then pressing the appropriate character. User-defined graphics will appear as normal letters until the program has been RUN.

*Hunchback, Artificial Intelligence,
Word Processor, utilities, snake games
and adventures.*

All this and much, much more
in an extra-large section of new programs
to key in and play.

MARATHON

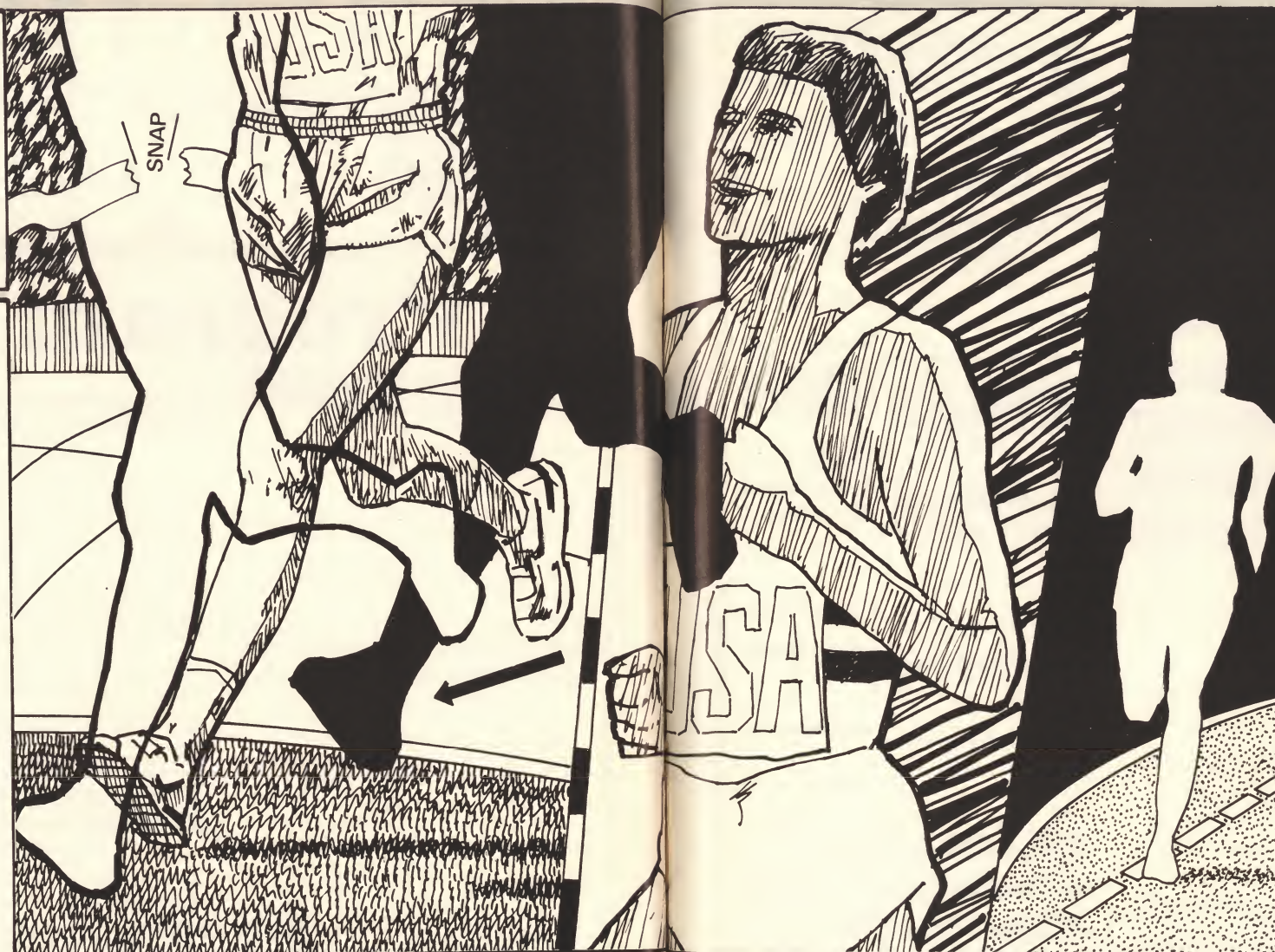
WITH EVERY muscle aching and your body rapidly drawing on reserves of fat and water it never knew it had, you are drawing to the close of a 26 mile marathon race.

P Bell of Houghton-Le-Spring in Tyne and Wear has written **Marathon** for the 16K ZX-81. You must decide how fast you want to run for each mile of the race, aiming for one of the coveted medals. But beware, if you go too fast too soon you will die of exhaustion before you ever reach the finishing line.

```

1 REM "MARATHON"
2 PRINT AT 0,0;"ENTER NAME"
3 INPUT U$
4 PRINT AT 1,0;"WHAT IS YOUR
BEST TIME, IF YOU DONOT HAVE ONE
ENTER 1000"
5 INPUT I
6 LET HS=I
7 PRINT AT 3,0;"ENTER HOW MANY
GOLD SILVER AND BRONZE MEDALS
YOU HAVE WON."
8 INPUT A
9 INPUT Z
10 INPUT C
11 LET G=A
12 LET S=Z
13 LET B=C
14 LET D=0
15 CLS
16 GOTO 1000
17 LET E=500
18 LET D=26
19 LET A=0
20 LET T=0
21 CLS
22 LET P=INT (RND*10)+1
23 IF P=7 OR P=8 OR P=9 OR P=10
THEN LET E=E-20
24 IF P=7 OR P=8 OR P=9 OR P=10
THEN LET T=T+3
25 IF P=4 OR P=5 OR P=6 THEN LET
E=E-10
26 IF P=4 OR P=5 OR P=6 THEN LET
T=T+1
27 IF E<0 THEN GOTO 200
28 PRINT AT 0,0;"DISTANCE TO GO
IS ";D;" MILE"
29 PRINT AT 2,0;"SPEED=";A;" MPH"
30 PRINT AT 4,0;"TIME=";T;" MINUTES"
31 PRINT AT 6,0;"ENERGY=";E;" UNITS"
32 PRINT AT 8,0;"POSITION=";P
33 IF P=2 THEN PRINT AT 8,11;"
ND"
34 IF P=3 THEN PRINT AT 8,11;"
RD"
35 IF P=4 OR P=5 OR P=6 OR P=7
OR P=8 OR P=9 OR P=10 THEN PRINT
AT 8,11;"TH"
36 IF P=1 THEN PRINT AT 8,11;"
ST"
37 PRINT AT 21,0;"ENTER SPEED:
15-151"
38 INPUT A
39 IF A<5 OR A>15 THEN GOTO 60
40 IF A=5 THEN LET E=E+10
41 IF A=6 THEN LET E=E+8
42 IF A=7 THEN LET E=E+6
43 IF A=8 THEN LET E=E+4
44 IF A=9 THEN LET E=E+2
45 IF A=10 THEN LET E=E-25
46 IF A=11 THEN LET E=E-30
47 IF A=12 THEN LET E=E-35
48 IF A=13 THEN LET E=E-40

```



```

94 IF A=14 THEN LET E=E-45
95 IF A=15 THEN LET E=E-50
96 IF A=5 THEN LET T=T+12
97 IF A=6 THEN LET T=T+10
98 IF A=7 THEN LET T=T+8.5
99 IF A=8 THEN LET T=T+7.5
100 IF A=9 THEN LET T=T+6.5
101 IF A=10 THEN LET T=T+5
102 IF A=11 THEN LET T=T+3.5
103 IF A=12 THEN LET T=T+3
104 IF A=13 THEN LET T=T+4.5
105 IF A=14 THEN LET T=T+4.2
106 IF A=15 THEN LET T=T+4
107 LET D=D-1
108 IF D=0 THEN GOTO 160
109 CLS
110 GOTO 29
111 CLS
112 LET R=INT (RND*10)+1
113 PRINT AT 0,0;"YOU FINISHED
IN A TIME OF ";AT 2,5;T;" MINUTE
S";AT 4,0;"THE WINNERS TIME WAS"
;AT 6,5;T-R;" MINUTES"
114 IF P=1 THEN PRINT AT 8,5;"
";AT 8,5;T;" MINUTES"
115 IF P=4 OR P=5 OR P=6 OR P=7
OR P=8 OR P=9 OR P=10 THEN PRINT
AT 10,26;"TH"
116 IF P=1 THEN PRINT AT 10,26;"
ST"
117 IF P=2 THEN PRINT AT 10,26;"
ND"

```

```

174 IF P=3 THEN PRINT AT 10,26;"
RD"
175 PRINT AT 10,0;"YOUR FINAL P
OSITION WAS ";P
176 IF P=1 THEN PRINT AT 13,0;"
YOU HAVE WON GOLD"
177 IF P=2 THEN PRINT AT 13,0;"
YOU HAVE WON SILVER"
178 IF P=3 THEN PRINT AT 13,0;"
YOU HAVE WON BRONZE"
179 IF HS>T THEN LET HS=T
180 PRINT AT 16,0;"YOUR REST TI
ME IS ";HS;" MINUTES"
181 IF P=1 THEN LET G=G+1
182 IF P=2 THEN LET S=S+1
183 IF P=3 THEN LET B=B+1
184 IF P=1 THEN PAUSE 300
185 IF P=2 THEN GOTO 5000
186 IF P=3 THEN PAUSE 30
187 IF P=2 OR P=3 THEN PAUSE 30
188 IF P=2 THEN GOTO 6000
189 IF P=3 THEN GOTO 7000
190 PRINT AT 21,0;"DO YOU WANT
ANOTHER GO (Y/N)"
191 INPUT B$
192 IF B$="Y" THEN GOTO 19
193 IF B$="N" THEN STOP
194 CLS
195 PRINT AT 0,0;"SORRY YOU HAV
E COLLAPSED AND HAVE FAILED T
O FINISH THE MARATHON"
196 PRINT AT 21,0;"PRESS P FOR

```

```

ANOTHER GO"
225 INPUT A$
226 IF A$<>"P" THEN STOP
230 IF A$="P" THEN GOTO 19
240 STOP
1000 CLS
1010 PRINT AT 0,0;"DO YOU WANT I
NSTRUCTIONS? (Y/N)"
1020 INPUT A$
1030 IF A$="Y" THEN GOTO 1100
1040 IF A$="N" THEN GOTO 19
1100 CLS
1110 PRINT AT 0,10;"MARATHON"
1120 PRINT AT 2,0;"YOU MUST COM
PLETE ALL 26 MILES IN AS QUICK A
S TIME AS POSSIBLE. THE FASTER YO
U RUN THE MORE ENERGY YOU US
E UP. DEPENDING ON THE POSITION
YOU ARE AT YOU MAY LOSE ENERGY A
ND TIME. YOU MUST FINISH THE 26
MILES WITH AT LEAST 1 UNIT
OF ENERGY LEFT. RUNNING AT 5
MPH YOU WILL GAIN ENERGY."
1130 PRINT AT 20,0;"PRESS R FOR
ANOTHER GO"
1140 INPUT B$
1150 IF B$="R" THEN GOTO 19
5000 CLS
5010 FOR A=0 TO 20
5020 PRINT "
5030 NEXT A
5040 PRINT AT 4,2;"THE WINNER OF
THE GOLD MEDAL";AT 6,2;"REPRESE
NTING";AT 8,2;"GREAT BRITAIN IS
";U$;AT 10,2;"IN A TIME OF ";T;"
MINUTES"
5041 PRINT AT 12,2;"YOU HAVE WON
";AT 14,2;G;"GOLD";AT 16,2;S;"
SILVER";AT 18,2;B;"BRONZE"
5050 PRINT AT 21,0;"PRESS C FOR
ANOTHER GO"
5055 INPUT F$
5060 IF F$="C" THEN GOTO 19
6000 CLS
6010 FOR A=0 TO 20
6020 PRINT "
6030 NEXT A
6040 PRINT AT 4,2;"THE WINNER OF
THE SILVER MEDAL";AT 6,2;"REPRE
SENTING";AT 8,2;"GREAT BRITAIN I
S ";U$;AT 10,2;"IN A TIME OF ";T
;" MINUTES"
6050 PRINT AT 12,2;"YOU HAVE WON
";AT 14,2;G;"GOLD";AT 16,2;S;"
SILVER";AT 18,2;B;"BRONZE"
6060 PRINT AT 21,0;"PRESS C FOR
ANOTHER GO"
6070 INPUT F$
6080 IF F$="C" THEN GOTO 19
7000 CLS
7010 PRINT AT 4,2;"THE WINNER OF
THE BRONZE MEDAL";AT 6,2;"REPRE
SENTING";AT 8,2;"GREAT BRITAIN I
S ";U$;AT 10,2;"IN A TIME OF ";T
;" MINUTES"
7020 PRINT AT 12,2;"YOU HAVE WON
";AT 14,2;G;"GOLD";AT 16,2;S;"
SILVER";AT 18,2;B;"BRONZE"
7030 PRINT AT 21,0;"PRESS C FOR
ANOTHER GO"
7040 INPUT F$
7050 IF F$="C" THEN GOTO 19

```



```

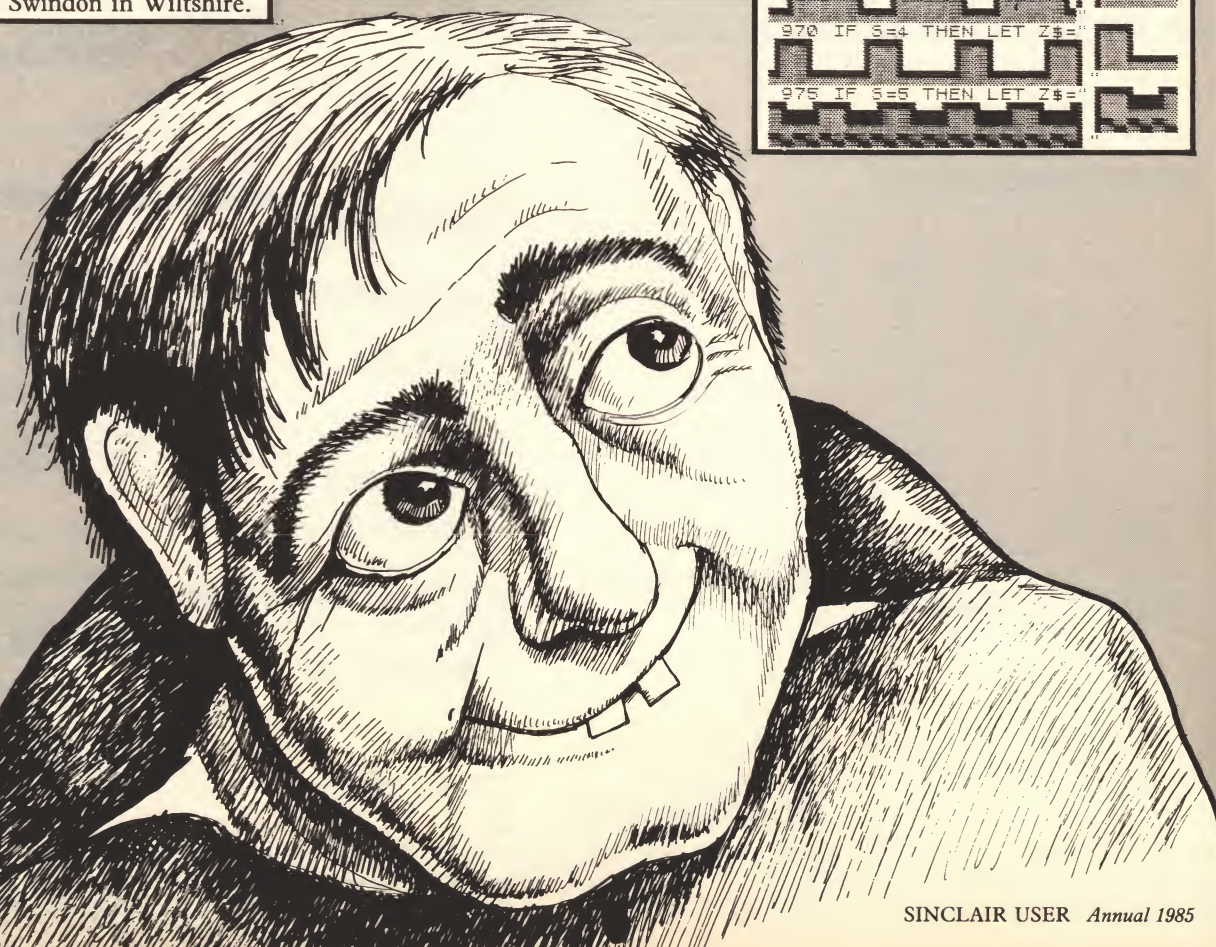
1 CLS
2 DIM A$(5,32)
3 LET A$(1)=""
4
5
6
7
8
9
10 LET A$(2)=""
11
12
13
14
15
16
17
18
19
20 LET A$(3)=""
21
22
23
24
25
26
27
28
29
30 LET A$(4)=""
31
32
33
34
35
36
37
38
39
40 LET A$(5)=""
41
42
43
44
45
46
47
48
49
50 DIM B$(6,16)
51 LET B$(1)=""
52
53
54
55
56
57
58
59
60 LET B$(2)=""
61
62
63
64
65
66
67
68
69
70 LET B$(3)=""
71
72
73
74
75
76
77
78
79
80 LET B$(4)=""
81
82
83
84
85
86
87
88
89
90 LET B$(5)=""
91
92
93
94
95
96
97
98
99
100 LET B$(6)=""
101
102
103
104
105
106
107
108
109
110 LET B$(7)=""
111
112
113
114
115
116
117
118
119
120 LET B$(8)=""
121
122
123
124
125
126
127
128
129
130 LET B$(9)=""
131
132
133
134
135
136
137
138
139
140 LET B$(10)=""
141
142
143
144
145
146
147
148
149
150 LET B$(11)=""
151
152
153
154
155
156
157
158
159
160 LET B$(12)=""
161
162
163
164
165
166
167
168
169
170 LET B$(13)=""
171
172
173
174
175
176
177
178
179
180 LET B$(14)=""
181
182
183
184
185
186
187
188
189
190 LET B$(15)=""
191
192
193
194
195
196
197
198
199
200 LET B$(16)=""
201
202
203
204
205
206
207
208
209
210 LET B$(17)=""
211
212
213
214
215
216
217
218
219
220 LET B$(18)=""
221
222
223
224
225
226
227
228
229
230 LET B$(19)=""
231
232
233
234
235
236
237
238
239
240 LET B$(20)=""
241
242
243
244
245
246
247
248
249
250 LET B$(21)=""
251
252
253
254
255
256
257
258
259
260 LET B$(22)=""
261
262
263
264
265
266
267
268
269
270 LET B$(23)=""
271
272
273
274
275
276
277
278
279
280 LET B$(24)=""
281
282
283
284
285
286
287
288
289
290 LET B$(25)=""
291
292
293
294
295
296
297
298
299
300 LET B$(26)=""
301
302
303
304
305
306
307
308
309
310 LET B$(27)=""
311
312
313
314
315
316
317
318
319
320 LET B$(28)=""
321
322
323
324
325
326
327
328
329
330 LET B$(29)=""
331
332
333
334
335
336
337
338
339
340 LET B$(30)=""
341
342
343
344
345
346
347
348
349
350 LET B$(31)=""
351
352
353
354
355
356
357
358
359
360 LET B$(32)=""
361
362
363
364
365
366
367
368
369
370 LET B$(33)=""
371
372
373
374
375
376
377
378
379
380 LET B$(34)=""
381
382
383
384
385
386
387
388
389
390 LET B$(35)=""
391
392
393
394
395
396
397
398
399
400 LET B$(36)=""
401
402
403
404
405
406
407
408
409
410 LET B$(37)=""
411
412
413
414
415
416
417
418
419
420 LET B$(38)=""
421
422
423
424
425
426
427
428
429
430 LET B$(39)=""
431
432
433
434
435
436
437
438
439
440 LET B$(40)=""
441
442
443
444
445
446
447
448
449
450 LET B$(41)=""
451
452
453
454
455
456
457
458
459
460 LET B$(42)=""
461
462
463
464
465
466
467
468
469
470 LET B$(43)=""
471
472
473
474
475
476
477
478
479
480 LET B$(44)=""
481
482
483
484
485
486
487
488
489
490 LET B$(45)=""
491
492
493
494
495
496
497
498
499
500 LET B$(46)=""
501
502
503
504
505
506
507
508
509
510 LET B$(47)=""
511
512
513
514
515
516
517
518
519
520 LET B$(48)=""
521
522
523
524
525
526
527
528
529
530 LET B$(49)=""
531
532
533
534
535
536
537
538
539
540 LET B$(50)=""
541
542
543
544
545
546
547
548
549
550 LET B$(51)=""
551
552
553
554
555
556
557
558
559
560 LET B$(52)=""
561
562
563
564
565
566
567
568
569
570 LET B$(53)=""
571
572
573
574
575
576
577
578
579
580 LET B$(54)=""
581
582
583
584
585
586
587
588
589
590 LET B$(55)=""
591
592
593
594
595
596
597
598
599
600 LET B$(56)=""
601
602
603
604
605
606
607
608
609
610 LET B$(57)=""
611
612
613
614
615
616
617
618
619
620 LET B$(58)=""
621
622
623
624
625
626
627
628
629
630 LET B$(59)=""
631
632
633
634
635
636
637
638
639
640 LET B$(60)=""
641
642
643
644
645
646
647
648
649
650 LET B$(61)=""
651
652
653
654
655
656
657
658
659
660 LET B$(62)=""
661
662
663
664
665
666
667
668
669
670 LET B$(63)=""
671
672
673
674
675
676
677
678
679
680 LET B$(64)=""
681
682
683
684
685
686
687
688
689
690 LET B$(65)=""
691
692
693
694
695
696
697
698
699
700 LET B$(66)=""
701
702
703
704
705
706
707
708
709
710 LET B$(67)=""
711
712
713
714
715
716
717
718
719
720 LET B$(68)=""
721
722
723
724
725
726
727
728
729
730 LET B$(69)=""
731
732
733
734
735
736
737
738
739
740 LET B$(70)=""
741
742
743
744
745
746
747
748
749
750 LET B$(71)=""
751
752
753
754
755
756
757
758
759
760 LET B$(72)=""
761
762
763
764
765
766
767
768
769
770 LET B$(73)=""
771
772
773
774
775
776
777
778
779
780 LET B$(74)=""
781
782
783
784
785
786
787
788
789
790 LET B$(75)=""
791
792
793
794
795
796
797
798
799
800 LET B$(76)=""
801
802
803
804
805
806
807
808
809
810 LET B$(77)=""
811
812
813
814
815
816
817
818
819
820 LET B$(78)=""
821
822
823
824
825
826
827
828
829
830 LET B$(79)=""
831
832
833
834
835
836
837
838
839
840 LET B$(80)=""
841
842
843
844
845
846
847
848
849
850 LET B$(81)=""
851
852
853
854
855
856
857
858
859
860 LET B$(82)=""
861
862
863
864
865
866
867
868
869
870 LET B$(83)=""
871
872
873
874
875
876
877
878
879
880 LET B$(84)=""
881
882
883
884
885
886
887
888
889
890 LET B$(85)=""
891
892
893
894
895
896
897
898
899
900 LET B$(86)=""
901
902
903
904
905
906
907
908
909
910 LET B$(87)=""
911
912
913
914
915
916
917
918
919
920 LET B$(88)=""
921
922
923
924
925
926
927
928
929
930 LET B$(89)=""
931
932
933
934
935
936
937
938
939
940 LET B$(90)=""
941
942
943
944
945
946
947
948
949
950 LET B$(91)=""
951
952
953
954
955
956
957
958
959
960 LET B$(92)=""
961
962
963
964
965
966
967
968
969
970 LET B$(93)=""
971
972
973
974
975
976
977
978
979
980 LET B$(94)=""
981
982
983
984
985
986
987
988
989
990 LET B$(95)=""
991
992
993
994
995
996
997
998
999
1000 LET B$(96)=""

```

QUASIMODO swings again in Hunchback for the 16K ZX-81. His girlfriend Esmeralda requires rescuing, but there are ten different battlements to cross before you reach her, and the soldiers will not be hanging about polishing their boots.

Full instructions are provided in the game, which was written by Paul Goodchild of Swindon in Wiltshire.

Hunchback



```

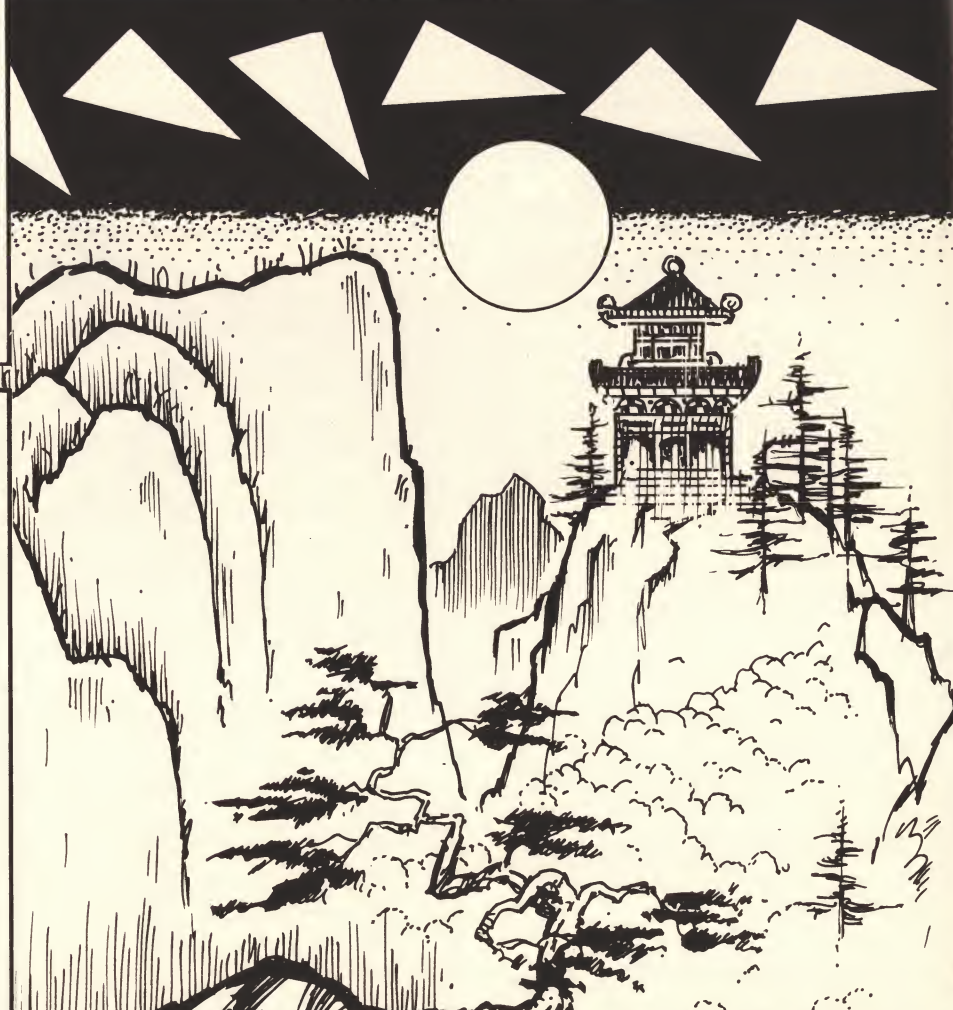
281 POKE 16418,0
282 FOR A=14 TO 23
283 PRINT AT A,0;"
284 NEXT A
285 PRINT AT 23,0;"
286 POKE 16418,2
287 IF INKEY="" THEN GOTO 290
288 POKE 16418,0
289 PRINT AT 0,0;"
290 FOR A=1 TO 12
291 PRINT AT A,0;"
292 NEXT A
293 PRINT AT 23,0;"
294 POKE 16418,2
295 IF INKEY="" THEN GOTO 298
296 POKE 16418,0
297 PRINT AT 0,0;"
298 FOR A=1 TO 12
299 PRINT AT A,0;"
300 NEXT A
301 PRINT AT 23,0;"
302 POKE 16418,2
303 IF INKEY="" THEN GOTO 306
304 POKE 16418,0
305 PRINT AT 0,0;"
306 FOR A=1 TO 12
307 PRINT AT A,0;"
308 NEXT A
309 PRINT AT 23,0;"
310 POKE 16418,2
311 IF INKEY="" THEN GOTO 314
312 POKE 16418,0
313 PRINT AT 0,0;"
314 FOR A=1 TO 12
315 PRINT AT A,0;"
316 NEXT A
317 PRINT AT 23,0;"
318 POKE 16418,2
319 IF INKEY="" THEN GOTO 322
320 POKE 16418,0
321 PRINT AT 0,0;"
322 FOR A=1 TO 12
323 PRINT AT A,0;"
324 NEXT A
325 PRINT AT 23,0;"
326 POKE 16418,2
327 IF INKEY="" THEN GOTO 330
328 POKE 16418,0
329 PRINT AT 0,0;"
330 FOR A=1 TO 12
331 PRINT AT A,0;"
332 NEXT A
333 PRINT AT 23,0;"
334 POKE 16418,2
335 IF INKEY="" THEN GOTO 338
336 POKE 16418,0
337 PRINT AT 0,0;"
338 FOR A=1 TO 12
339 PRINT AT A,0;"
340 NEXT A
341 PRINT AT 23,0;"
342 POKE 16418,2
343 IF INKEY="" THEN GOTO 346
344 POKE 16418,0
345 PRINT AT 0,0;"
346 FOR A=1 TO 12
347 PRINT AT A,0;"
348 NEXT A
349 PRINT AT 23,0;"
350 POKE 16418,2
351 IF INKEY="" THEN GOTO 354
352 POKE 16418,0
353 PRINT AT 0,0;"
354 FOR A=1 TO 12
355 PRINT AT A,0;"
356 NEXT A
357 PRINT AT 23,0;"
358 POKE 16418,2
359 IF INKEY="" THEN GOTO 362
360 POKE 16418,0
361 PRINT AT 0,0;"
362 FOR A=1 TO 12
363 PRINT AT A,0;"
364 NEXT A
365 PRINT AT 23,0;"
366 POKE 16418,2
367 IF INKEY="" THEN GOTO 370
368 POKE 16418,0
369 PRINT AT 0,0;"
370 FOR A=1 TO 12
371 PRINT AT A,0;"
372 NEXT A
373 PRINT AT 23,0;"
374 POKE 16418,2
375 IF INKEY="" THEN GOTO 378
376 POKE 16418,0
377 PRINT AT 0,0;"
378 FOR A=1 TO 12
379 PRINT AT A,0;"
380 NEXT A
381 PRINT AT 23,0;"
382 POKE 16418,2
383 IF INKEY="" THEN GOTO 386
384 POKE 16418,0
385 PRINT AT 0,0;"
386 FOR A=1 TO 12
387 PRINT AT A,0;"
388 NEXT A
389 PRINT AT 23,0;"
390 POKE 16418,2
391 IF INKEY="" THEN GOTO 394
392 POKE 16418,0
393 PRINT AT 0,0;"
394 FOR A=1 TO 12
395 PRINT AT A,0;"
396 NEXT A
397 PRINT AT 23,0;"
398 POKE 16418,2
399 IF INKEY="" THEN GOTO 402
400 POKE 16418,0
401 PRINT AT 0,0;"
402 FOR A=1 TO 12
403 PRINT AT A,0;"
404 NEXT A
405 PRINT AT 23,0;"
406 POKE 16418,2
407 IF INKEY="" THEN GOTO 410
408 POKE 16418,0
409 PRINT AT 0,0;"
410 FOR A=1 TO 12
411 PRINT AT A,0;"
412 NEXT A
413 PRINT AT 23,0;"
414 POKE 16418,2
415 IF INKEY="" THEN GOTO 418
416 POKE 16418,0
417 PRINT AT 0,0;"
418 FOR A=1 TO 12
419 PRINT AT A,0;"
420 NEXT A
421 PRINT AT 23,0;"
422 POKE 16418,2
423 IF INKEY="" THEN GOTO 426
424 POKE 16418,0
425 PRINT AT 0,0;"
426 FOR A=1 TO 12
427 PRINT AT A,0;"
428 NEXT A
429 PRINT AT 23,0;"
430 POKE 16418,2
431 IF INKEY="" THEN GOTO 434
432 POKE 16418,0
433 PRINT AT 0,0;"
434 FOR A=1 TO 12
435 PRINT AT A,0;"
436 NEXT A
437 PRINT AT 23,0;"
438 POKE 16418,2
439 IF INKEY="" THEN GOTO 442
440 POKE 16418,0
441 PRINT AT 0,0;"
442 FOR A=1 TO 12
443 PRINT AT A,0;"
444 NEXT A
445 PRINT AT 23,0;"
446 POKE 16418,2
447 IF INKEY="" THEN GOTO 450
448 POKE 16418,0
449 PRINT AT 0,0;"
450 FOR A=1 TO 12
451 PRINT AT A,0;"
452 NEXT A
453 PRINT AT 23,0;"
454 POKE 16418,2
455 IF INKEY="" THEN GOTO 458
456 POKE 16418,0
457 PRINT AT 0,0;"
458 FOR A=1 TO 12
459 PRINT AT A,0;"
460 NEXT A
461 PRINT AT 23,0;"
462 POKE 16418,2
463 IF INKEY="" THEN GOTO 466
464 POKE 16418,0
465 PRINT AT 0,0;"
466 FOR A=1 TO 12
467 PRINT AT A,0;"
468 NEXT A
469 PRINT AT 23,0;"
470 POKE 16418,2
471 IF INKEY="" THEN GOTO 474
472 POKE 16418,0
473 PRINT AT 0,0;"
474 FOR A=1 TO 12
475 PRINT AT A,0;"
476 NEXT A
477 PRINT AT 23,0;"
478 POKE 16418,2
479 IF INKEY="" THEN GOTO 482
480 POKE 16418,0
481 PRINT AT 0,0;"
482 FOR A=1 TO 12
483 PRINT AT A,0;"
484 NEXT A
485 PRINT AT 23,0;"
486 POKE 16418,2
487 IF INKEY="" THEN GOTO 490
488 POKE 16418,0
489 PRINT AT 0,0;"
490 FOR A=1 TO 12
491 PRINT AT A,0;"
492 NEXT A
493 PRINT AT 23,0;"
494 POKE 16418,2
495 IF INKEY="" THEN GOTO 498
496 POKE 16418,0
497 PRINT AT 0,0;"
498 FOR A=1 TO 12
499 PRINT AT A,0;"
500 NEXT A
501 PRINT AT 23,0;"
502 POKE 16418,2
503 IF INKEY="" THEN GOTO 506
504 POKE 16418,0
505 PRINT AT 0,0;"
506 FOR A=1 TO 12
507 PRINT AT A,0;"
508 NEXT A
509 PRINT AT 23,0;"
510 POKE 16418,2
511 IF INKEY="" THEN GOTO 514
512 POKE 16418,0
513 PRINT AT 0,0;"
514 FOR A=1 TO 12
515 PRINT AT A,0;"
516 NEXT A
517 PRINT AT 23,0;"
518 POKE 16418,2
519 IF INKEY="" THEN GOTO 522
520 POKE 16418,0
521 PRINT AT 0,0;"
522 FOR A=1 TO 12
523 PRINT AT A,0;"
524 NEXT A
525 PRINT AT 23,0;"
526 POKE 16418,2
527 IF INKEY="" THEN GOTO 530
528 POKE 16418,0
529 PRINT AT 0,0;"
530 FOR A=1 TO 12
531 PRINT AT A,0;"
532 NEXT A
533 PRINT AT 23,0;"
534 POKE 16418,2
535 IF INKEY="" THEN GOTO 538
536 POKE 16418,0
537 PRINT AT 0,0;"
538 FOR A=1 TO 12
539 PRINT AT A,0;"
540 NEXT A
541 PRINT AT 23,0;"
542 POKE 16418,2
543 IF INKEY="" THEN GOTO 546
544 POKE 16418,0
545 PRINT AT 0,0;"
546 FOR A=1 TO 12
547 PRINT AT A,0;"
548 NEXT A
549 PRINT AT 23,0;"
550 POKE 16418,2
551 IF INKEY="" THEN GOTO 554
552 POKE 16418,0
553 PRINT AT 0,0;"
554 FOR A=1 TO 12
555 PRINT AT A,0;"
556 NEXT A
557 PRINT AT 23,0;"
558 POKE 16418,2
559 IF INKEY="" THEN GOTO 562
560 POKE 16418,0
561 PRINT AT 0,0;"
562 FOR A=1 TO 12
563 PRINT AT A,0;"
564 NEXT A
565 PRINT AT 23,0;"
566 POKE 16418,2
567 IF INKEY="" THEN GOTO 570
568 POKE 16418,0
569 PRINT AT 0,0;"
570 FOR A=1 TO 12
571 PRINT AT A,0;"
572 NEXT A
573 PRINT AT 23,0;"
574 POKE 16418,2
575 IF INKEY="" THEN GOTO 578
576 POKE 16418,0
577 PRINT AT 0,0;"
578 FOR A=1 TO 12
579 PRINT AT A,0;"
580 NEXT A
581 PRINT AT 23,0;"
582 POKE 16418,2
583 IF INKEY="" THEN GOTO 586
584 POKE 16418,0
585 PRINT AT 0,0;"
586 FOR A=1 TO 12
587 PRINT AT A,0;"
588 NEXT A
589 PRINT AT 23,0;"
590 POKE 16418,2
591 IF INKEY="" THEN GOTO 594
592 POKE 16418,0
593 PRINT AT 0,0;"
594 FOR A=1 TO 12
595 PRINT AT A,0;"
596 NEXT A
597 PRINT AT 23,0;"
598 POKE 16418,2
599 IF INKEY="" THEN GOTO 602
600 POKE 16418,0
601 PRINT AT 0,0;"
602 FOR A=1 TO 12
603 PRINT AT A,0;"
604 NEXT A
605 PRINT AT 23,0;"
606 POKE 16418,2
607 IF INKEY="" THEN GOTO 610
608 POKE 16418,0
609 PRINT AT 0,0;"
610 FOR A=1 TO 12
611 PRINT AT A,0;"
612 NEXT A
613 PRINT AT 23,0;"
614 POKE 16418,2
615 IF INKEY="" THEN GOTO 618
616 POKE 16418,0
617 PRINT AT 0,0;"
618 FOR A=1 TO 12
619 PRINT AT A,0;"
620 NEXT A
621 PRINT AT 23,0;"
622 POKE 16418,2
623 IF INKEY="" THEN GOTO 626
624 POKE 16418,0
625 PRINT AT 0,0;"
626 FOR A=1 TO 12
627 PRINT AT A,0;"
628 NEXT A
629 PRINT AT 23,0;"
630 POKE 16418,2
631 IF INKEY="" THEN GOTO 634
632 POKE 16418,0
633 PRINT AT 0,0;"
634 FOR A=1 TO 12
635 PRINT AT A,0;"
636 NEXT A
637 PRINT AT 23,0;"
638 POKE 16418,2
639 IF INKEY="" THEN GOTO 642
640 POKE 16418,0
641 PRINT AT 0,0;"
642 FOR A=1 TO 12
643 PRINT AT A,0;"
644 NEXT A
645 PRINT AT 23,0;"
646 POKE 16418,2
647 IF INKEY="" THEN GOTO 650
648 POKE 16418,0
649 PRINT AT 0,0;"
650 FOR A=1 TO 12
651 PRINT AT A,0;"
652 NEXT A
653 PRINT AT 23,0;"
654 POKE 16418,2
655 IF INKEY="" THEN GOTO 658
656 POKE 16418,0
657 PRINT AT 0,0;"
658 FOR A=1 TO 12
659 PRINT AT A,0;"
660 NEXT A
661 PRINT AT 23,0;"
662 POKE 16418,2
663 IF INKEY="" THEN GOTO 666
664 POKE 16418,0
665 PRINT AT 0,0;"
666 FOR A=1 TO 12
667 PRINT AT A,0;"
668 NEXT A
669 PRINT AT 23,0;"
670 POKE 16418,2
671 IF INKEY="" THEN GOTO 674
672 POKE 16418,0
673 PRINT AT 0,0;"
674 FOR A=1 TO 12
675 PRINT AT A,0;"
676 NEXT A
677 PRINT AT 23,0;"
678 POKE 16418,2
679 IF INKEY="" THEN GOTO 682
680 POKE 16418,0
681 PRINT AT 0,0;"
682 FOR A=1 TO 12
683 PRINT AT A,0;"
684 NEXT A
685 PRINT AT 23,0;"
686 POKE 16418,2
687 IF INKEY="" THEN GOTO 690
688 POKE 16418,0
689 PRINT AT 0,0;"
690 FOR A=1 TO 12
691 PRINT AT A,0;"
692 NEXT A
693 PRINT AT 23,0;"
694 POKE 16418,2
695 IF INKEY="" THEN GOTO 698
696 POKE 16418,0
697 PRINT AT 0,0;"
698 FOR A=1 TO 12
699 PRINT AT A,0;"
700 NEXT A
701 PRINT AT 23,0;"
702 POKE 16418,2
703 IF INKEY="" THEN GOTO 706
704 POKE 16418,0
705 PRINT AT 0,0;"
706 FOR A=1 TO 12
707 PRINT AT A,0;"
708 NEXT A
709 PRINT AT 23,0;"
710 POKE 16418,2
711 IF INKEY="" THEN GOTO 714
712 POKE 16418,0
713 PRINT AT 0,0;"
714 FOR A=1 TO 12
715 PRINT AT A,0;"
716 NEXT A
717 PRINT AT 23,0;"
718 POKE 16418,2
719 IF INKEY="" THEN GOTO 722
720 POKE 16418,0
721 PRINT AT 0,0;"
722 FOR A=1 TO 12
723 PRINT AT A,0;"
724 NEXT A
725 PRINT AT 23,0;"
726 POKE 16418,2
727 IF INKEY="" THEN GOTO 730
728 POKE 16418,0
729 PRINT AT 0,0;"
730 FOR A=1 TO 12
731 PRINT AT A,0;"
732 NEXT A
733 PRINT AT 23,0;"
734 POKE 16418,2
735 IF INKEY="" THEN GOTO 738
736 POKE 16418,0
737 PRINT AT 0,0;"
738 FOR A=1 TO 12
739 PRINT AT A,0;"
740 NEXT A
741 PRINT AT 23,0;"
742 POKE 16418,2
743 IF INKEY="" THEN GOTO 746
744 POKE 16418,0
745 PRINT AT 0,0;"
746 FOR A=1 TO 12
747 PRINT AT A,0;"
748 NEXT A
749 PRINT AT 23,0;"
750 POKE 16418,2
751 IF INKEY="" THEN GOTO 754
752 POKE 16418,0
753 PRINT AT 0,0;"
754 FOR A=1 TO 12
755 PRINT AT A,0;"
756 NEXT A
757 PRINT AT 23,0;"
758 POKE 16418,2
759 IF INKEY="" THEN GOTO 762
760 POKE 16418,0
761 PRINT AT 0,0;"
762 FOR A=1 TO 12
763 PRINT AT A,0;"
764 NEXT A
765 PRINT AT 23,0;"
766 POKE 16418,2
767 IF INKEY="" THEN GOTO 770
768 POKE 16418,0
769 PRINT AT 0,0;"
770 FOR A=1 TO 12
771 PRINT AT A,0;"
772 NEXT A
773 PRINT AT 23,0;"
774 POKE 16418,2
775 IF INKEY="" THEN GOTO 778
776 POKE 16418,0
777 PRINT AT 0,0;"
778 FOR A=1 TO 12
779 PRINT AT A,0;"
780 NEXT A
781 PRINT AT 23,0;"
782 POKE 16418,2
783 IF INKEY="" THEN GOTO 786
784 POKE 16418,0
785 PRINT AT 0,0;"
786 FOR A=1 TO 12
787 PRINT AT A,0;"
788 NEXT A
789 PRINT AT 23,0;"
790 POKE 16418,2
791 IF INKEY="" THEN GOTO 794
792 POKE 16418,0
793 PRINT AT 0,0;"
794 FOR A=1 TO 12
795 PRINT AT A,0;"
796 NEXT A
797 PRINT AT 23,0;"
798 POKE 16418,2
799 IF INKEY="" THEN GOTO 802
800 POKE 16418,0
801 PRINT AT 0,0;"
802 FOR A=1 TO 12
803 PRINT AT A,0;"
804 NEXT A
805 PRINT AT 23,0;"
806 POKE 16418,2
807 IF INKEY="" THEN GOTO 810
808 POKE 16418,0
809 PRINT AT 0,0;"
810 FOR A=1 TO 12
811 PRINT AT A,0;"
812 NEXT A
813 PRINT AT 23,0;"
814 POKE 16418,2
815 IF INKEY="" THEN GOTO 818
816 POKE 16418,0
817 PRINT AT 0,0;"
818 FOR A=1 TO 12
819 PRINT AT A,0;"
820 NEXT A
821 PRINT AT 23,0;"
822 POKE 16418,2
823 IF INKEY="" THEN GOTO 826
824 POKE 16418,0
825 PRINT AT 0,0;"
826 FOR A=1 TO 12
827 PRINT AT A,0;"
828 NEXT A
829 PRINT AT 23,0;"
830 POKE 16418,2
831 IF INKEY="" THEN GOTO 834
832 POKE 16418,0
833 PRINT AT 0,0;"
834 FOR A=1 TO 12
835 PRINT AT A,0;"
836 NEXT A
837 PRINT AT 23,0;"
838 POKE 16418,2
839 IF INKEY="" THEN GOTO 842
840 POKE 16418,0
841 PRINT AT 0,0;"
842 FOR A=1 TO 12
843 PRINT AT A,0;"
844 NEXT A
845 PRINT AT 23,0;"
846 POKE 16418,2
847 IF INKEY="" THEN GOTO 850
848 POKE 16418,0
849 PRINT AT 0,0;"
850 FOR A=1 TO 12
851 PRINT AT A,0;"
852 NEXT A
853 PRINT AT 23,0;"
854 POKE 16418,2
855 IF INKEY="" THEN GOTO 858
856 POKE 16418,0
857 PRINT AT 0,0;"
858 FOR A=1 TO 12
859 PRINT AT A,0;"
860 NEXT A
861 PRINT AT 23,0;"
862 POKE 16418,2
863 IF INKEY="" THEN GOTO 866
864 POKE 16418,0
865 PRINT AT 0,0;"
866 FOR A=1 TO 12
867 PRINT AT A,0;"
868 NEXT A
869 PRINT AT 23,0;"
870 POKE 16418,2
871 IF INKEY="" THEN GOTO 874
872 POKE 16418,0
873 PRINT AT 0,0;"
874 FOR A=1 TO 12
875 PRINT AT A,0;"
876 NEXT A
877 PRINT AT 23,0;"
878 POKE 16418,2
879 IF INKEY="" THEN GOTO 882
880 POKE 16418,0
881 PRINT AT 0,0;"
882 FOR A=1 TO 12
883 PRINT AT A,0;"
884 NEXT A
885 PRINT AT 23,0;"
886 POKE 16418,2
887 IF INKEY="" THEN GOTO 890
888 POKE 16418,0
889 PRINT AT 0,0;"
890 FOR A=1 TO 12
891 PRINT AT A,0;"
892 NEXT A
893 PRINT AT 23,0;"
894 POKE 16418,2
895 IF INKEY="" THEN GOTO 898
896 POKE 16418,0
897 PRINT AT 0,0;"
898 FOR A=1 TO 12
899 PRINT AT A,0;"
900 NEXT A
901 PRINT AT 23,0;"
902 POKE 16418,2
903 IF INKEY="" THEN GOTO 906
904 POKE 16418,0
905 PRINT AT 0,0;"
906 FOR A=1 TO 12
907 PRINT AT A,0;"
908 NEXT A
909 PRINT AT 23,0;"
910 POKE 16418,2
911 IF INKEY="" THEN GOTO 914
912 POKE 16418,0
913 PRINT AT 0
```


THE SOFT MUSIC of tinkling stones and the peaceful beauty of the Japanese Garden are threatened by a mechanical bird, a symbol of evil. You must deprive the bird of its weapon and defeat it in order to save the island, but you will probably have more trouble staying alive, for the island is full of mysterious objects. M A Phillips wrote this text adventure for the 16K ZX-81.

```

REM COPYRIGHT M A P
LET HSC=0
LET C=0
LET SA=0
GOSUB 1800
GOSUB 1500
GOSUB 9900
LET S=0*(R0)
IF V$=0*(INT (RND*15)+1)
IF V$=0*(8) AND P=1 THEN GO
TO 15
20 IF C=5 AND P=1 THEN GOTO 87
23 PRINT AT 17,0;"YOU CAN SEE
";V$
GOSUB 9900
PRINT AT 17,0;"YOU ARE ";S$
GOSUB 9900
IF R0=0 THEN GOSUB 400
GOTO 65
FAST
GOSUB 9900
FAST
PRINT AT 17,0;"YOU HAVE:--"
FOR F=1 TO 3
PRINT C$(F)
NEXT F
SLOW
GOSUB 9900
GOSUB 9900
PRINT AT 17,0;"COMMAND:"
71 IF C=7 THEN PRINT AT 18,0;
"YOU'RE GETTING WEAK"
76 INPUT A$
78 LET C=C+1
77 IF C=10 THEN GOTO 9600
GOSUB 9900
IF A$="I" THEN GOTO 1000
IF A$="J" THEN GOTO 40
IF A$="K" THEN GOSUB 1500
IF A$="L" THEN GOTO 2000
IF A$="M" THEN GOTO 1900
IF A$="N" THEN GOTO 4000
IF A$="O" THEN GOTO 6000
IF A$="P" THEN GOTO 7000
IF A$="Q" THEN GOTO 8000
IF A$="R" THEN GOTO 9700
200 PRINT AT 4,3;"A NOBLENT 6TO
A BREAKS OUT AT 9,3;YOU ARE
AT 14,7;END
210 PAUSE 300
215 LET C=C-9
230 GOTO 10
410 IF INT (RND*3)=0 THEN RETUR
N
411 GOSUB 9900
412 PRINT AT 17,0;"SOMETHING BL
ACK IS MOVING TOWARDS YOU
..."
413 PAUSE 90
414 GOSUB 9900
415 IF P=0 THEN GOTO 450
420 PRINT AT 17,0;"IT HAS SENSE
D THAT YOU HAVE THE H A T E AN
D SCORES YOU OFF..."
422 FOR N=0 TO 40
424 NEXT N
425 GOTO 1900
430 GOTO 8900
450 PRINT AT 17,0;"YOU ARE VERY
STILL... AND IT HAS
GONE!"
451 LET C=C+1
454 PAUSE 90
457 GOSUB 9900
458 PRINT AT 17,9;"
460 RETURN
1000 IF C$(1,1 TO 7)<N$ AND C$(
2,1 TO 7)<N$ AND C$(3,1 TO 7)<
N$ THEN GOTO 2600
1001 IF V$=0*(10) THEN PRINT AT
14,0;"DONT BE DAFT..."
1002 IF V$=0*(2) OR V$=0*(5) OR
V$=0*(7) OR V$=0*(10) THEN GOTO
20
1003 IF V$<0*(5) THEN GOTO 1007
1004 PRINT AT 14,0;"SHE SLAPS YO
U"
1005 LET C=C+2
1006 GOTO 60

```



JAPANESE GARDEN

```

1007 IF V$=0*(9) OR V$=0*(4) OR
V$=0*(13) THEN PRINT AT 14,0;"YO
UR SCORE HAS BEEN DECREASED"
1008 IF V$=0*(12) OR V$=0*(11) T
HEN LET S=3+75
1009 IF V$=0*(12) OR V$=0*(11) T
HEN PRINT AT 14,0;"YOUR SCORE HA
S BEEN INCREASED"
1010 IF V$=0*(1) THEN LET SA=1
1015 IF C$(1,1 TO 7)=N$ THEN GOT
O 1030
1020 IF C$(2,1 TO 7)=N$ THEN GOT
O 1040
1025 IF C$(3,1 TO 7)=N$ THEN GOT
O 1050
1030 LET C$(1)=V$
1035 GOTO 1070
1040 LET C$(2)=V$
1045 GOTO 1070
1050 LET C$(3)=V$
1071 IF V$=0*(3) THEN LET C=C-5
1072 PRINT AT 17,0;"O.K. WITHDR
AW"
1073 IF V$=0*(8) THEN LET S=3+50
1074 IF V$=0*(8) THEN LET P=1
1075 GOSUB 9900
1077 LET V$=N$
1080 GOTO 20
1500 FAST
1510 CLS
1520 SLOW
1522 PRINT AT 1,8;"JAPANESE GARD
EN"
1523 PRINT TAB 8;"
1530 PRINT AT 5,12;"MOVE";TAB
12;"-----";TAB 1;"F:FORWARD;B:B
ACKWARD;S:SIDEWAYS"
1534 PRINT AT 9,9;"TAKE"
1535 PRINT TAB 9;"DROP"
1536 PRINT TAB 9;"REST"
1540 PRINT TAB 9;"SAKE (DRIN
K)"
1542 PRINT TAB 9;"FLOWER (SNIF
F)"
1543 PRINT TAB 9;"INVENTORY"
1544 PRINT TAB 9;"HELP"
1545 PRINT TAB 9;"PROTECTION"
1546 PRINT TAB 9;"KEY PAGE (THI
S)"
1547 PRINT TAB 9;"QUIT GAME"
1548 PRINT AT 21,5;"HIT A KEY TO
CONTINUE"
1549 IF INKEY$<" " THEN GOTO 154
8
1550 IF INKEY$=" " THEN GOTO 1550
1550 CLS
1550 GOSUB 9900
1700 RETURN
1800 FAST
1802 CLS
1803 FOR N=20 TO 1 STEP -1
1805 PRINT AT N,0;"
1806 NEXT N
1807 SLOW
1808 PRINT AT 0,6;"JAPANESE GA
RDEN"
1809 IF INKEY$="J" THEN RETURN
1810 PRINT AT 1,1;"YOU'RE SEARCH
ING A PEACEFUL JAPANESE GARDEN"
1811 IF INKEY$="J" THEN RETURN
1812 PRINT AT 5,1;"YOU MUST FIND
A WEAPON TO DEFEAT THE MECHANICAL
BIRD"
1813 PRINT AT 1,1;"YOUR WEAPON IS
A METAL BIRD"
1814 PRINT AT 12,1;"THE METAL
BIRD IS A WEAPON THAT YOU CAN USE
TO DEFEAT THE MECHANICAL BIRD"
1815 IF INKEY$="J" THEN RETURN
1816 PRINT AT 1,1;"YOUR WEAPON IS
A METAL BIRD"
1817 PRINT AT 1,1;"YOUR WEAPON IS
A METAL BIRD"
1818 PRINT AT 20,19;"GOOD LUCK"
1820 IF INKEY$="J" THEN RETURN
1824 PRINT AT 0,6;"PRESS "

```



```

1825 PRINT AT 18,8;"YOU SAIL AWAY"
1826 FOR N=0 TO 38
1827 NEXT N
1828 PRINT AT 18,8;"THE METAL
BIRD"
1829 FOR N=0 TO 40
1830 NEXT N
1831 PRINT AT 18,8;"L I V
E"
1832 FOR G=1 TO 3
1833 FOR N=0 TO 30
1834 NEXT N
1835 FOR N=2 TO 30 STEP 2
1837 PRINT AT 1,N-2;"
1838 PRINT AT 1,N;"U"
1841 NEXT N
1842 PRINT AT 1,30;"
1843 PRINT AT 18,8;"
1890 NEXT G
1892 FAST
1894 CLS
1896 SLOW
1899 GOTO 9820
2005 PRINT AT 17,0;"WHICH ITEM?
(1,2 OR 3)"
2006 INPUT DR
2010 IF DR<1 OR DR>3 THEN GOTO 2
010
2011 IF C$(DR)<0*(1) THEN GOTO
1925
2012 FOR N=0 TO 20
2013 NEXT N
2013 PRINT AT 17,0;"YOU CAN'T.
"
2014 PAUSE 100
2015 GOTO 2000
2016 IF C$(DR)=0*(3) THEN LET S=
3-5
2017 IF C$(DR)=0*(14) THEN LET C
=C+2
2018 PRINT AT 17,0;"YOU DROPPED
"
2019 IF C$(DR)=0*(14) THEN PRINT
"THE METAL SPEAR HAS SHATTERED
BLINDING YOU BADLY..."
2020 IF C$(DR)=0*(12) THEN PRINT
"THE MACHINE HAS EXPLODED AND
HURT YOU..."
2021 IF C$(DR)=0*(12) THEN LET C
=C+2
2022 IF C$(DR)=0*(8) THEN LET S=
3-100
2023 IF C$(DR)=0*(8) THEN LET P=
0
2024 IF C$(DR)<0*(15) THEN GOTO
2400
2025 PRINT "THE BOX EMITS A SCRE
ECHING TONE THE EVIL-ONE AWAKES
"
2100 FOR N=0 TO 50
2120 NEXT N
2130 GOTO 8700
2400 LET C$(DR)=N$
2500 GOSUB 9900
2510 GOTO 20
2500 PRINT AT 16,0;"TRY DROPPING
SOMETHING FIRST."
2505 SCROLL
2510 GOTO 20
3005 FAST
3010 CLS
3015 SLOW
3018 GOTO 9820
4035 LET S=3+50
4037 IF P=0 THEN LET S=3-30
4040 GOTO 8700
5003 IF SA=0 THEN GOTO 5200
5005 IF V$<0*(2) AND V$<0*(5)
AND V$<0*(7) THEN GOTO 5100
5020 IF INT (RND*5)=2 THEN GOTO
5300
5024 PRINT AT 14,0;"O.K."
5025 PRINT AT 15,0;"YOU ARE REFR
ESHED"
5027 LET C=C-4
5028 PRINT AT 17,0;"WITHDRAWAL
"
5030 LET V$=N$
5032 LET S=3+10
5033 GOSUB 9900
5035 GOTO 20
5100 PRINT AT 14,0;"YOU CAN'T DR
INK HERE"
5110 GOTO 20
5200 PRINT AT 14,0;"YOU'VE NO SA
KE"
5220 GOTO 20
5300 PRINT AT 17,0;"THE SAKE IS
POISONOUS: 1 TO 6
5305 FOR F=1 TO 6
5310 GOSUB 9900
5315 NEXT F
5320 GOTO 8900
6005 IF INT (RND*3)=1 THEN GOTO
6500
6010 PRINT AT 17,0;"THIS IS A LI
FE-PLANT"
6012 LET C=C-5
6015 GOSUB 9900
6016 PRINT AT 17,0;"WITHDRAWAL
"
6018 LET C=C-5
6020 GOSUB 9900
6022 LET V$=N$
6025 GOTO 20
6500 PRINT AT 17,0;"THIS FLOWER
IS DEATH..."
6505 FOR F=1 TO 6
6510 GOSUB 9900
6515 NEXT F
6520 GOTO 8900
7001 IF V$<0*(2) AND V$<0*(5)
AND V$<0*(7) AND V$<0*(8) THEN
GOTO 7100
7005 IF INT (RND*3)=1 THEN GOTO
7500
7008 IF SA=1 AND C=0 THEN GOTO 5
300
7010 PRINT AT 14,0;"O.K."
7011 PRINT AT 15,0;"YOU WAKE UP
SOMEWHERE ELSE"
7012 LET C=C-5
7013 PRINT AT 17,0;"WITHDRAWAL
"
7015 GOSUB 9900
7016 LET RO=RO+2
7025 IF RO>27 THEN LET RO=1
7030 GOTO 10
7100 PRINT AT 14,0;"NO RESTING H
ERE"
7110 GOTO 20
7500 PRINT AT 17,0;"YOU ENTER A
CALM SLEEP"
7505 GOSUB 9900
7510 PRINT AT 17,0;"AND WAKE UP
SNIPPING A FLOWER"
7520 GOSUB 9900
8000 IF C<-10 THEN GOTO 8005
8001 PRINT AT 14,0;"YOU'VE FOUND
A DINGY..."
8002 PAUSE 100
8003 GOTO 1900
8005 PRINT AT 17,0;"F*B*3"
8010 INPUT Z$
8012 IF RO=2 AND P=0 THEN GOTO 8
700
8015 IF Z$=" " THEN LET RO=RO+1
8025 IF RO<1 OR RO>27 OR Z$="S"
THEN LET RO=INT (RND*27)+1
8027 LET S=3+1
8028 GOSUB 9900
8030 GOTO 10
8700 FAST
8701 CLS
8702 FOR N=0 TO 21
8705 PRINT AT N,0;"
8710 NEXT N
8712 SLOW
8714 IF C=5 AND P=1 AND RO<2 TH
EN GOTO 200
8715 PRINT AT 2,6;"YOU HAVE ANGE
"
8720 PRINT AT 4,9;"METAL BIRD"
8725 FOR N=0 TO 50
8728 NEXT N
8729 PRINT AT 2,6;"
8730 PRINT AT 19,1;"SUT SU H
"
8731 IF P=1 AND RO<2 THEN PRINT
AT 19,1;"H A T E"
8732 IF P=1 AND RO=2 THEN PRINT
AT 19,1;"H A T E"
8733 IF P=1 AND RO=2 THEN LET S=
3+100
8734 FOR N=0 TO 40
8736 NEXT N
8738 PRINT AT 19,1;"TO PRO
"
8740 FOR N=0 TO 50
8742 NEXT N
8745 PRINT AT 19,1;"
8750 PRINT AT 10,7;"-1- S L O
"
8752 FOR N=0 TO 30

```

```

5020 IF INT (RND*5)=2 THEN GOTO
5300
5024 PRINT AT 14,0;"O.K."
5025 PRINT AT 15,0;"YOU ARE REFR
ESHED"
5027 LET C=C-4
5028 PRINT AT 17,0;"WITHDRAWAL
"
5030 LET V$=N$
5032 LET S=3+10
5033 GOSUB 9900
5035 GOTO 20
5100 PRINT AT 14,0;"YOU CAN'T DR
INK HERE"
5110 GOTO 20
5200 PRINT AT 14,0;"YOU'VE NO SA
KE"
5220 GOTO 20
5300 PRINT AT 17,0;"THE SAKE IS
POISONOUS: 1 TO 6
5305 FOR F=1 TO 6
5310 GOSUB 9900
5315 NEXT F
5320 GOTO 8900
6005 IF INT (RND*3)=1 THEN GOTO
6500
6010 PRINT AT 17,0;"THIS IS A LI
FE-PLANT"
6012 LET C=C-5
6015 GOSUB 9900
6016 PRINT AT 17,0;"WITHDRAWAL
"
6018 LET C=C-5
6020 GOSUB 9900
6022 LET V$=N$
6025 GOTO 20
6500 PRINT AT 17,0;"THIS FLOWER
IS DEATH..."
6505 FOR F=1 TO 6
6510 GOSUB 9900
6515 NEXT F
6520 GOTO 8900
7001 IF V$<0*(2) AND V$<0*(5)
AND V$<0*(7) AND V$<0*(8) THEN
GOTO 7100
7005 IF INT (RND*3)=1 THEN GOTO
7500
7008 IF SA=1 AND C=0 THEN GOTO 5
300
7010 PRINT AT 14,0;"O.K."
7011 PRINT AT 15,0;"YOU WAKE UP
SOMEWHERE ELSE"
7012 LET C=C-5
7013 PRINT AT 17,0;"WITHDRAWAL
"
7015 GOSUB 9900
7016 LET RO=RO+2
7025 IF RO>27 THEN LET RO=1
7030 GOTO 10
7100 PRINT AT 14,0;"NO RESTING H
ERE"
7110 GOTO 20
7500 PRINT AT 17,0;"YOU ENTER A
CALM SLEEP"
7505 GOSUB 9900
7510 PRINT AT 17,0;"AND WAKE UP
SNIPPING A FLOWER"
7520 GOSUB 9900
8000 IF C<-10 THEN GOTO 8005
8001 PRINT AT 14,0;"YOU'VE FOUND
A DINGY..."
8002 PAUSE 100
8003 GOTO 1900
8005 PRINT AT 17,0;"F*B*3"
8010 INPUT Z$
8012 IF RO=2 AND P=0 THEN GOTO 8
700
8015 IF Z$=" " THEN LET RO=RO+1
8025 IF RO<1 OR RO>27 OR Z$="S"
THEN LET RO=INT (RND*27)+1
8027 LET S=3+1
8028 GOSUB 9900
8030 GOTO 10
8700 FAST
8701 CLS
8702 FOR N=0 TO 21
8705 PRINT AT N,0;"
8710 NEXT N
8712 SLOW
8714 IF C=5 AND P=1 AND RO<2 TH
EN GOTO 200
8715 PRINT AT 2,6;"YOU HAVE ANGE
"
8720 PRINT AT 4,9;"METAL BIRD"
8725 FOR N=0 TO 50
8728 NEXT N
8729 PRINT AT 2,6;"
8730 PRINT AT 19,1;"SUT SU H
"
8731 IF P=1 AND RO<2 THEN PRINT
AT 19,1;"H A T E"
8732 IF P=1 AND RO=2 THEN PRINT
AT 19,1;"H A T E"
8733 IF P=1 AND RO=2 THEN LET S=
3+100
8734 FOR N=0 TO 40
8736 NEXT N
8738 PRINT AT 19,1;"TO PRO
"
8740 FOR N=0 TO 50
8742 NEXT N
8745 PRINT AT 19,1;"
8750 PRINT AT 10,7;"-1- S L O
"
8752 FOR N=0 TO 30

```



```

8754 NEXT N
8755 FAST
8756 CLS
8757 SLOW
8758 PRINT AT 17,14;"0";TAB 14;"
8759 TAB 14;"
8760 FOR N=0 TO 30
8761 NEXT N
8762 PRINT AT 1,28;" "AT 2,27;"
8763 AT 1,28;" "AT 2,27;"
8764 PRINT AT 4,26;" "AT 2,27;"
8770 PRINT AT 7,23;" "AT 4,26;"
8772 PRINT AT 7,23;" "AT 12,1;"
8773 PRINT AT 12,16;" "AT 15,8;"
8774 IF R0=2 AND P=1 THEN GOTO 9
8775 PRINT AT 16,8;"
8777 PRINT AT 17,14;"
8778 PRINT AT 12,6;"
8779 PRINT AT 12,6;"
8780 PRINT AT 4,1;"AT 7,3;"
8781 PRINT AT 8,0;"
8782 FOR N=0 TO 20
8783 NEXT N
8784 PRINT AT 6,4;"I T R I P P
8785 O F F"
8791 PRINT TAB 4;"

```

```

8792 PRINT AT 9,6;"Y O U R H E
8793 A D"
8795 PRINT TAB 6;"

```

```

8820 FOR N=0 TO 30
8821 NEXT N
8822 PRINT AT 18,13;"TAB 13;"
8823 AT 18,12;"TAB 12;"AT 19
8824 PRINT AT 17,14;"TAB 14;"
8825 TAB 14;"
8826 FOR N=0 TO 30
8827 NEXT N
8828 GOTO 8820
8829 GOTO 8820
8830 FOR N=0 TO 20
8831 NEXT N
8832 GOTO 3000
8833 FAST
8834 DIM D$(27,36)
8835 LET D$(1)="LISTENING TO A S
8836OUND SCULPTURE"
8837 LET D$(2)="WITH THE METAL
8838BIRD"
8839 LET D$(3)="IN A DESERTED CA
8840PARK"
8841 LET D$(4)="IN A VIDEO ARCADE
8842"
8843 LET D$(5)="IN A SHOPPING CO
8844MPLEX"
8845 LET D$(6)="SAT UNDER A COOD
8846NUT PALM"
8847 LET D$(7)="BY A TALL OFFICE
8848BLOCK"
8849 LET D$(8)="CLOSE TO A VOLCA
8850NO"
8851 LET D$(9)="ON A SPIRAL ESC
8852LATOR"
8853 LET D$(10)="IN A DISUSED RA
8854ILURARY TUNNEL"
8855 LET D$(11)="BY AN ORNAMENTA
8856L POOL"
8857 LET D$(12)="INSIDE A GIANT
8858DOOME"
8859 LET D$(13)="BY A GOLDEN LAK
8860E"
8861 LET D$(14)="SITTING ON A SU
8862NNY SLOPE"
8863 LET D$(15)="IN A THICKET OF
8864BUSH"
8865 LET D$(16)="ON A VERMILLION
8866BEACH"
8867 LET D$(17)="LISTENING TO A
8868SCULPTURE"
8869 LET D$(18)="WITH THE SOUND
8870SCULPTURE"
8871 LET D$(19)="IN A L O N G C
8872ORRADOR"
8873 LET D$(20)="BY A RICE FIELD
8874"
8875 LET D$(21)="LOST"
8876 LET D$(22)="NEAR A GENTLY-F
8877LOWING RIVER"
8878 LET D$(23)="IN A LASER FORE
8879ST"
8880 LET D$(24)="LEARNING AGAINST
8881A WILLOW"
8882 LET D$(25)="IN A ROBOTIC HO
8883PITAL"
8884 LET D$(26)="IN A TUBE-SHAPE
8885D LIFT"
8886 LET D$(27)="NEXT TO A GLIDE
8887"
8888 DIM O$(15,17)
8889 LET O$(1)="SOME SAKE"
8890 LET O$(2)="A NEON-GARDEN"
8891 LET O$(3)="A BLUE-ROCK"
8892 LET O$(4)="A GLASS SWORD"
8893 LET O$(5)="A NEON-PARK"
8894 LET O$(6)="A JAPANESE GIRL"
8895 LET O$(7)="A NEON-ROCKERY"
8896 LET O$(8)="THE P P P P P"
8897 LET O$(9)="A SILVER DRB"
8898 LET O$(10)="A BIRD IN THE S
8899KY"
8900 LET O$(11)="A GREEN BOOK"
8901 LET O$(12)="A PINK MACHINE"
8902 LET O$(13)="A DRAGON KITE"
8903 LET O$(14)="A LAGBIT SPEAR"
8904 LET O$(15)="A BLACK BOX"
8905 LET S=0
8906 LET N$="NOTHING"
8907 DIM C$(3,17)

```



```

9230 FOR F=1 TO 3
9231 LET C$(F)=N$
9232 NEXT F
9233 LET P=0
9234 LET R0=INT (RND*27)+1
9235 SLOW
9236 RETURN
9237 FOR N=1 TO 21 STEP 2
9238 PRINT AT N,0;"E X H A
9239R"
9240 NEXT N
9241 FAST
9242 CLS
9243 SLOW
9244 GOTO 1900
9245 IF INT (RND*5)=1 THEN GOTO
9246705
9247 IF INT (RND*5)=2 THEN GOTO
9248710
9249 IF INT (RND*5)=0 THEN GOTO
9250720
9251 IF INT (RND*5)=4 THEN GOTO
9252730
9253 IF INT (RND*5)=3 THEN GOTO
9254740
9255 PRINT "HELP MINERALS ARE
9256HEALTHY."
9257 SCROLL
9258 GOTO 20
9259 PRINT "HELP BIRDS LIKE MU
9260IC"
9261 SCROLL
9262 GOTO 20
9263 SCROLL
9264 PRINT "HELP NEONS ARE MAG
9265IC"
9266 SCROLL
9267 GOTO 20
9268 PRINT "HELP WINE IS THE B
9269EST REST."
9270 SCROLL
9271 GOTO 20
9272 PRINT "HELP THE WEAPON IS
9273NIT NICE."
9274 SCROLL
9275 GOTO 20
9276 PRINT AT 12,8;"
9277 AT 16,8;"
9284 PRINT AT 8,8;"
9285 AT 12,8;"

```

```

9807 PRINT AT 4,8;"
9808 AT 8,8;"
9809 AT 4,8;"
9810 PRINT AT 2,4;"AT 13,8;"
9811 AT 7,26;"AT 1,24;"
9812 PRINT AT 14,19;"
9813 PRINT AT 8,3;"AT 16,25;"
9814 AT 12,14;"
9815 IF S<0 THEN LET S=0
9816 PRINT AT 5,10;"SCORE:";S*10
9817 IF S<HSC THEN GOTO 9830
9818 FOR N=1 TO 10
9819 PRINT AT 0,0;"YOU HAVE
9820THE HI-SCORE"
9821 AT 0,0;"
9822 NEXT N
9823 PRINT "INPUT YOUR N
9824AME:"
9825 INPUT W$
9826 IF S=HSC THEN LET HSC=S
9827 PRINT AT 20,9;"HI-SCORE:";H
9828 SC*10;AT 21,9;"BY:";W$
9829 PRINT AT 13,8;"ANOTHER TRY"
9830 IF INKEY$="N" THEN STOP
9831 PRINT AT 13,8;"
9832 IF INKEY$="Y" THEN GOTO 995
9833
9834 GOTO 9836
9835 FAST
9836 SCROLL
9837 SCROLL
9838 SCROLL
9839 SCROLL
9840 SLOW
9841 RETURN
9842 FAST
9843 CLS
9844 SLOW
9845 GOTO 3
9846 SLOW
9847 GOTO 1
9848 CLEAR
9849 SAVE "JAPANESE GARDEN"
9850 GOTO 9990

```

monty

```

1 PRINT AT 21,2; FLASH 1;"STO
P TAPE AND PRESS ANY KEY": PAUSE
0: CLS
5 LET bs=0: LET b$="no one"
25 POKE 23609,100
27 GO SUB 1000
30 GO SUB 500
35 LET gc=0: DIM s(10): DIM s$
(10,3)
40 LET sh=1: LET i=0: LET p=7:
INK i: PAPER p: BORDER 6: CLS
45 BORDER 6: PAPER 6
50 FOR n=0 TO 31: PRINT AT 0,n
;"#";AT 21,n;"#": NEXT n
60 FOR n=0 TO 21: PRINT AT n,3
1;"#";AT n,0;"#": NEXT n: PAPER
P
70 LET foodc=2: LET rock=1
80 LET snake=4: LET food=p*8+f
oodc
100 LET land=p*8+i: LET mx=1: L
ET my=0
110 LET s=1: LET x=2: LET y=8
120 DIM x(200): DIM y(200)
130 FOR n=1 TO s
140 LET i$=INKEY$

```

MONTY is hungry in his little field, so when food drops out of the sky he is delighted. Unfortunately rocks come tumbling down as well and if he hits any he will die. As Monty eats he grows, so watch out or he will snap himself up by mistake.

Monty Python was written by Guy Morgan of Llantrisant in Glamorgan, for any Spectrum.

The game uses our special abbreviations for graphics characters, so please read the instructions on the first page of Program Printout.

```

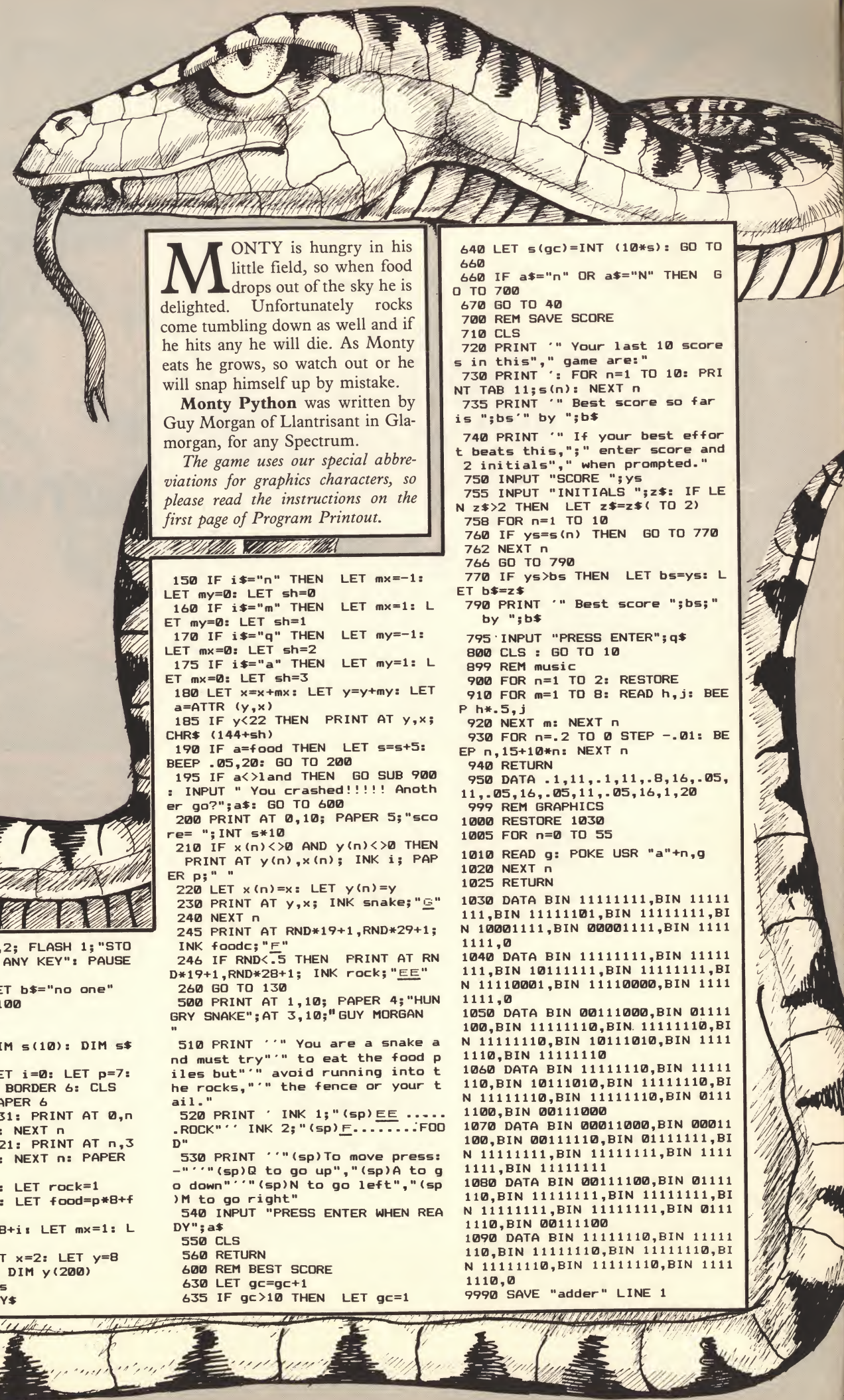
150 IF i$="n" THEN LET mx=-1:
LET my=0: LET sh=0
160 IF i$="m" THEN LET mx=1: L
ET my=0: LET sh=1
170 IF i$="q" THEN LET my=-1:
LET mx=0: LET sh=2
175 IF i$="a" THEN LET my=1: L
ET mx=0: LET sh=3
180 LET x=x+mx: LET y=y+my: LET
a=ATTR (y,x)
185 IF y<22 THEN PRINT AT y,x;
CHR$ (144+sh)
190 IF a=food THEN LET s=s+5:
BEEP .05,20: GO TO 200
195 IF a<>land THEN GO SUB 900
: INPUT "You crashed!!!! Anoth
er go?";a$: GO TO 600
200 PRINT AT 0,10; PAPER 5;"sco
re=";INT s*10
210 IF x(n)<>0 AND y(n)<>0 THEN
PRINT AT y(n),x(n); INK i; PAP
ER p;"
220 LET x(n)=x: LET y(n)=y
230 PRINT AT y,x; INK snake;"C"
240 NEXT n
245 PRINT AT RND*19+1,RND*29+1;
INK foodc;"E"
246 IF RND<.5 THEN PRINT AT RN
D*19+1,RND*28+1; INK rock;"EE"
260 GO TO 130
500 PRINT AT 1,10; PAPER 4;"HUN
GRY SNAKE";AT 3,10;"GUY MORGAN"
"
510 PRINT " "You are a snake a
nd must try" "to eat the food p
iles but" "avoid running into t
he rocks," "the fence or your t
ail."
520 PRINT " INK 1;"(sp)EE .....
.ROCK" INK 2;"(sp)E.....FOO
D"
530 PRINT " "(sp)To move press:
-" "(sp)Q to go up," "(sp)A to g
o down" "(sp)N to go left," "(sp
)M to go right"
540 INPUT "PRESS ENTER WHEN REA
DY";a$
550 CLS
560 RETURN
600 REM BEST SCORE
630 LET gc=gc+1
635 IF gc>10 THEN LET gc=1

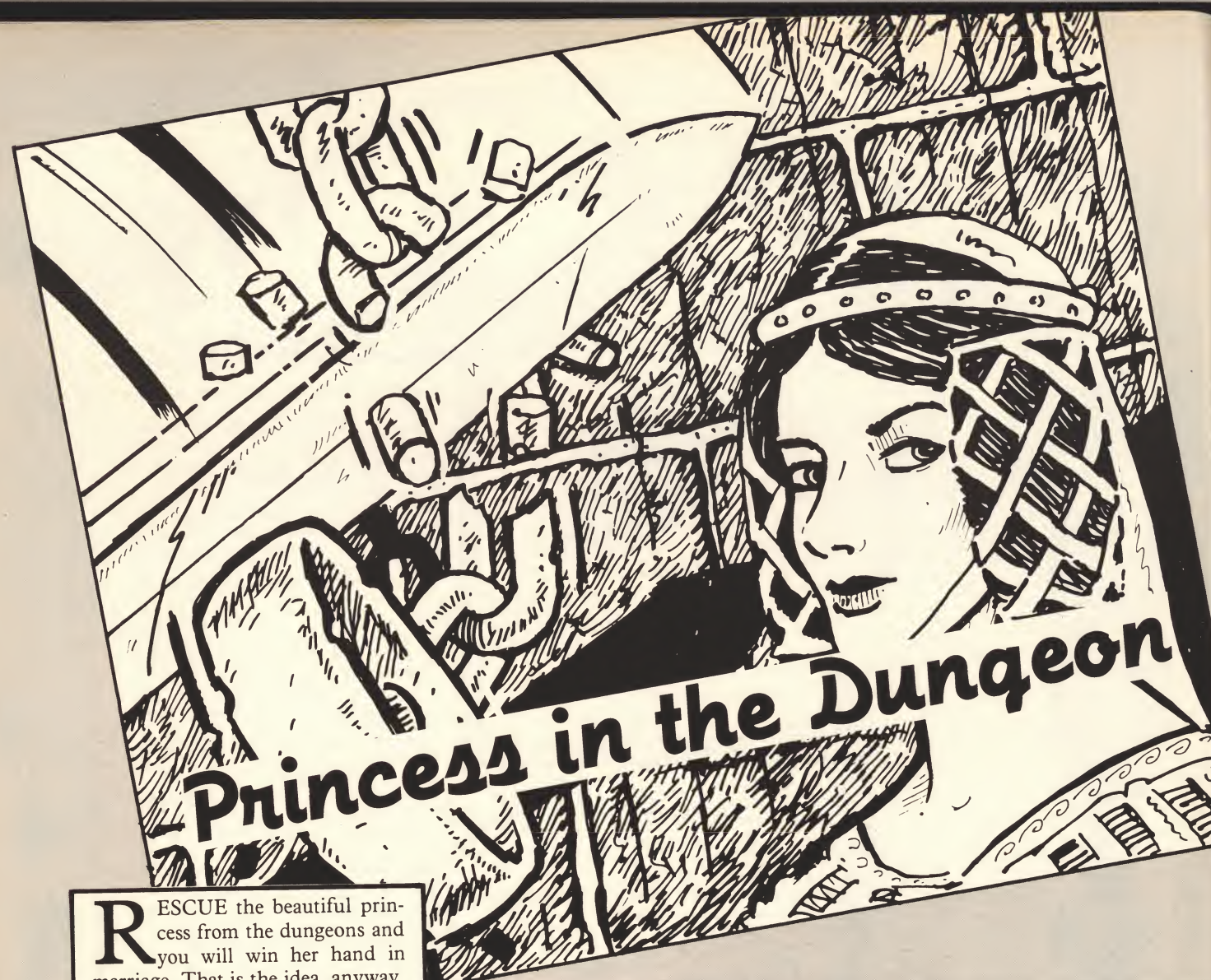
```

```

640 LET s(gc)=INT (10*s): GO TO
660
660 IF a$="n" OR a$="N" THEN G
O TO 700
670 GO TO 40
700 REM SAVE SCORE
710 CLS
720 PRINT "Your last 10 score
s in this," game are:"
730 PRINT "FOR n=1 TO 10: PRI
NT TAB 11;s(n): NEXT n
735 PRINT "Best score so far
is ";bs" by ";b$
740 PRINT "If your best effor
t beats this," enter score and
2 initials," when prompted."
750 INPUT "SCORE ";ys
755 INPUT "INITIALS ";z$: IF LE
N z$>2 THEN LET z$=z$( TO 2)
758 FOR n=1 TO 10
760 IF ys=s(n) THEN GO TO 770
762 NEXT n
766 GO TO 790
770 IF ys>bs THEN LET bs=ys: L
ET b$=z$
790 PRINT "Best score ";bs;"
by ";b$
795 INPUT "PRESS ENTER";q$
800 CLS : GO TO 10
899 REM music
900 FOR n=1 TO 2: RESTORE
910 FOR m=1 TO 8: READ h,j: BEE
P h*.5,j
920 NEXT m: NEXT n
930 FOR n=.2 TO 0 STEP -.01: BE
EP n,15+10*n: NEXT n
940 RETURN
950 DATA 1,11,1,11,.8,16,.05,
11,.05,16,.05,11,.05,16,1,20
999 REM GRAPHICS
1000 RESTORE 1030
1005 FOR n=0 TO 55
1010 READ g: POKE USR "a"+n,g
1020 NEXT n
1025 RETURN
1030 DATA BIN 11111111,BIN 11111
111,BIN 11111101,BIN 11111111,BI
N 10001111,BIN 00001111,BIN 1111
1111,0
1040 DATA BIN 11111111,BIN 11111
111,BIN 10111111,BIN 11111111,BI
N 11110001,BIN 11110000,BIN 1111
1111,0
1050 DATA BIN 00111000,BIN 01111
100,BIN 11111110,BIN 11111110,BI
N 11111110,BIN 10111010,BIN 1111
1110,BIN 11111110
1060 DATA BIN 11111110,BIN 11111
110,BIN 10111010,BIN 11111110,BI
N 11111111,BIN 11111111,BIN 0111
1100,BIN 00111000
1070 DATA BIN 00011000,BIN 00011
100,BIN 00111110,BIN 01111111,BI
N 11111111,BIN 11111111,BIN 1111
1111,BIN 11111111
1080 DATA BIN 00111100,BIN 01111
110,BIN 11111111,BIN 11111111,BI
N 11111111,BIN 11111110,BIN 0111
1110,BIN 00111100
1090 DATA BIN 11111110,BIN 11111
110,BIN 11111110,BIN 11111110,BI
N 11111110,BIN 11111110,BIN 1111
1110,0
9990 SAVE "adder" LINE 1

```





Princess in the Dungeon

RESCUE the beautiful princess from the dungeons and you will win her hand in marriage. That is the idea, anyway, but you will have to take the scenic route to reach the maiden in the game by Neil Martin of Fareham in Hampshire for the 16K Spectrum.

A simple text adventure game, ideal for introducing newcomers to the genre, the game accepts two word commands such as GET ROPE or KILL MAN. Movement is accomplished simply by typing NORTH or WEST. Make sure you are in CAPS LOCK mode before playing the game, however, or the program will not understand you at all.

9000 BORDER 0: PAPER 0: INK 5
10000 LET G=0: LET H\$="NOTHING":
LET N=1: LET H=0: LET M=1: LET K
=0: LET J=1: LET W=0: LET L=0: L
ET I=1: LET B=0: LET F=1: LET D=
1: LET S=0: LET E=1: LET R=0: LE

```
T A=1: LET C=0
1010 LET W$="WEST": LET E$="EAST"
": LET N$="NORTH": LET S$="SOUTH"
"
1020 GO TO 1090
1030 PRINT "'YOU ARE ";a$
1040 PRINT "EXITS LEAD ";b$
1050 PRINT "YOU SEE ";c$
1060 INPUT "WHAT NOW ? ";A$
1070 PRINT A$
1080 RETURN
1090 LET a$="IN CASTLE ZOLAN"
1100 IF A=1 THEN LET c$="A ROPE"
"
1110 IF A=0 THEN LET c$=H$
1120 LET b$="SOUTH,WEST"
1130 GO SUB 1030
1140 IF G=1 THEN PRINT "WELL DO
NE YOU HAVE RESCUED THE PRINCES
S !!!": PRINT "THE KING ALLOWS
YOU TO MARRY HER": PRINT "YOU L
IVE HAPPILY EVER AFTER !!!": GO TO
2260
1150 IF A=S$ THEN GO TO 1800
1160 IF A=W$ THEN GO TO 1280
1170 IF A$="GET ROPE" AND A=1 TH
EN LET R=1: LET A=0: GO SUB 231
0: GO TO 1090
1180 GO SUB 2230: GO TO 1090
1190 LET a$="STANDING IN BASTON
TOWN"
1200 IF E=1 THEN LET c$="A SHOR
T SWORD"
1210 IF E=0 THEN LET c$=H$
1220 LET b$="EAST,SOUTH"
1230 GO SUB 1030
1240 IF A=E$ THEN GO TO 1280
1250 IF A$="GET SWORD" AND E=1 T
HEN LET S=1: LET E=0: GO SUB 23
10: GO TO 1190
1260 IF A=S$ THEN GO TO 1480
1270 GO SUB 2230: GO TO 1190
```

```
1280 LET a$="STANDING BENEATH MO
UNT SNOW"
1290 LET c$="A LEDGE ABOVE YOU"
1300 LET b$=W$
1310 GO SUB 1030
1320 IF A=W$ THEN GO TO 1190
1330 IF A$="THROW ROPE" AND R=1
THEN LET O=INT (RND*3): GO TO 1
360
1340 IF A$="CLIMB ROPE" AND R=2
THEN LET P=INT (RND*3): GO TO 1
380
1350 GO SUB 2230: GO TO 1280
1360 IF O=1 THEN PRINT "THE ROP
E DOES NOT CATCH,IT FALLSBACK IN
TO YOUR ARMS": GO TO 1280
1370 IF O=0 OR O=2 THEN PRINT "
THE ROPE CATCHES IN THE BRANCHES
OF A TREE": LET R=2: GO TO 1280
1380 IF P=1 THEN PRINT "YOU STA
RT TO CLIMB,BUT YOU FALL ...": G
O TO 2250
1390 IF P=0 OR P=2 THEN PRINT "
YOU CLIMB UP THE ROPE": GO TO 14
00
1400 LET a$="STANDING ON THE LED
GE"
1410 IF D=1 THEN LET c$="A GOLD
COIN"
1420 IF D=0 THEN LET c$=H$
1430 LET b$="DOWN"
1440 GO SUB 1030
1450 IF A$="DOWN" THEN GO TO 12
80
1460 IF A$="GET COIN" AND D=1 TH
EN LET C=1: LET D=0: GO SUB 231
0: GO TO 1400
1470 GO SUB 2230: GO TO 1400
1480 LET a$="IN THE TOWN OF WEST
PORT"
1490 LET b$=S$
1500 IF F=1 THEN LET c$="A MAN
```

```
TRYING TO SELL A BOAT"
1510 IF F=0 THEN LET c$=H$
1520 GO SUB 1030
1530 IF A=S$ THEN GO TO 1560
1540 IF A$="BUY BOAT" AND C=1 TH
EN PRINT "YOU BUY THE BOAT": LE
T B=1: LET F=0: GO TO 1480
1550 GO SUB 2230: GO TO 1480
1560 LET a$="AT THE CLEAR RIVER"
1570 LET c$="A WIDE RIVER EAST"
1580 LET b$=N$
1590 GO SUB 1030
1600 IF A=N$ THEN GO TO 1480
1610 IF A$="ROW BOAT" AND B=1 TH
EN PRINT "YOU ROW ACROSS THE CL
EAR RIVER": GO TO 1630
1620 GO SUB 2230: GO TO 1560
1630 LET a$="WALKING ALONG A NAR
ROW PATH "
1640 IF I=1 THEN LET c$="A LONG
LADDER"
1650 IF I=0 THEN LET c$=H$
1660 LET b$=E$
1670 GO SUB 1030
1680 IF A=E$ THEN GO TO 1710
1690 IF A$="GET LADDER" AND I=1
THEN LET L=1: LET I=0: GO SUB 2
310: GO TO 1630
1700 GO SUB 2230: GO TO 1630
1710 LET a$="BY THE GREAT WALL"
1720 LET c$="A TALL WALL SOUTH"
1730 LET b$="WEST,NORTH"
1740 GO SUB 1030
1750 IF A=W$ THEN GO TO 1630
1760 IF A=N$ THEN GO TO 1800
1770 IF A$="LEAN LADDER" AND L=1
AND W=0 THEN PRINT "YOU LEAN T
HE LADDER AGAINST THE WALL": LET
W=1: GO TO 1710
1780 IF A$="CLIMB LADDER" AND W=
1 THEN PRINT "YOU CLIMB OVER TH
E WALL": GO TO 1890
```

```
1790 GO SUB 2230: GO TO 1710
1800 LET a$="STANDING BY A LAKE"
1810 LET c$="A BRIDGE GOING SOUT
H"
1820 LET b$="NORTH,SOUTH"
1830 GO SUB 1030
1840 IF A=N$ THEN GO TO 1090
1850 IF A=S$ THEN LET T=INT (R
ND*2): GO TO 1870
1860 GO SUB 2230: GO TO 1800
1870 IF T=0 THEN PRINT "YOU CRO
SS THE BRIDGE SAFELY": GO TO 171
0
1880 IF T=1 THEN PRINT "YOU STA
RT TO CROSS,BUT YOU FALL ...": G
O TO 2250
1890 LET a$="OUTSIDE THE TOWER O
F DOOM"
1900 IF J=0 THEN LET c$=H$
1910 IF J=1 THEN LET c$="A KEY"
1920 LET b$="NORTH,WEST"
1930 GO SUB 1030
1940 IF A=N$ THEN GO TO 1710
1950 IF A=W$ THEN GO TO 2060
1960 IF A$="GET KEY" AND J=1 THE
N LET J=0: LET K=1: GO SUB 2310
: GO TO 1890
1970 GO SUB 2230: GO TO 1890
1980 LET a$="IN THE ARMOUR ROOM"
1990 IF M=0 THEN LET c$=H$
2000 IF M=1 THEN LET c$="A SHIE
LD"
2010 LET b$=E$
2020 GO SUB 1030
2030 IF A=E$ THEN GO TO 2060
2040 IF A$="GET SHIELD" AND M=1
THEN LET H=1: LET M=0: GO SUB 2
310: GO TO 1980
2050 GO SUB 2230: GO TO 1980
2060 LET a$="OUTSIDE A DUNGEON"
2070 IF N=0 THEN LET c$="THE DU
NGEON DOOR,IT HAS A KEYHOLE"
```

```
2080 IF N=1 THEN LET c$="A GUAR
D OUTSIDE THE DUNGEON"
2090 LET b$="EAST,WEST"
2100 GO SUB 1030
2110 IF A=W$ THEN GO TO 1980
2120 IF A$="KILL GUARD" AND S=1
AND H=1 AND N=1 THEN PRINT "YOU
KILL THE GUARD": LET N=0: GO TO
2060
2130 IF A=E$ THEN GO TO 1890
2140 IF A$="KILL GUARD" AND S=0
OR H=0 AND N=1 THEN PRINT "THE
GUARD ATTACKS YOU...": GO TO 225
0
2150 IF A$="INSERT KEY" AND K=1
AND N=0 THEN PRINT "YOU OPEN TH
E DOOR AND GO INSIDE": GO TO 217
0
2160 GO SUB 2230: GO TO 2060
2170 LET a$="IN THE DUNGEON"
2180 LET c$="THE PRINCESS"
2190 LET b$="OUT"
2200 GO SUB 1030
2210 IF A$="OUT" THEN LET G=1:
PRINT "YOU GO OUT AND THE PRINCE
SS FOLLOWS": GO TO 2060
2220 GO SUB 2230: GO TO 2170
2230 IF A$=N$ OR A$=S$ OR A$=E$
OR A$=W$ THEN PRINT "YOU CAN'T
GO ";A$: RETURN
2240 PRINT "I DON'T UNDERSTAND":
RETURN
2250 PRINT "" YOU ARE D
EAD !!!"
2260 PRINT "" DO YOU WANT TO PL
AY AGAIN ? Y/N"
2270 INPUT A$
2280 IF A$="Y" THEN CLS : GO TO
1000
2290 IF A$="N" THEN STOP
2300 GO TO 2270
2310 PRINT "YOU PICK IT UP": RET
URN
```


DOGS WILL LOVE this program, by R P Banks of Leigh-on-Sea in Essex. You must guide your faithful hound around a maze, picking up dog biscuits before the time runs out. But the dog catchers are out to get him, so you must plan carefully to evade them.

Use keys 2 and W to move up and down, and O and P to move left and right. Key H holds the game, and key A quits.

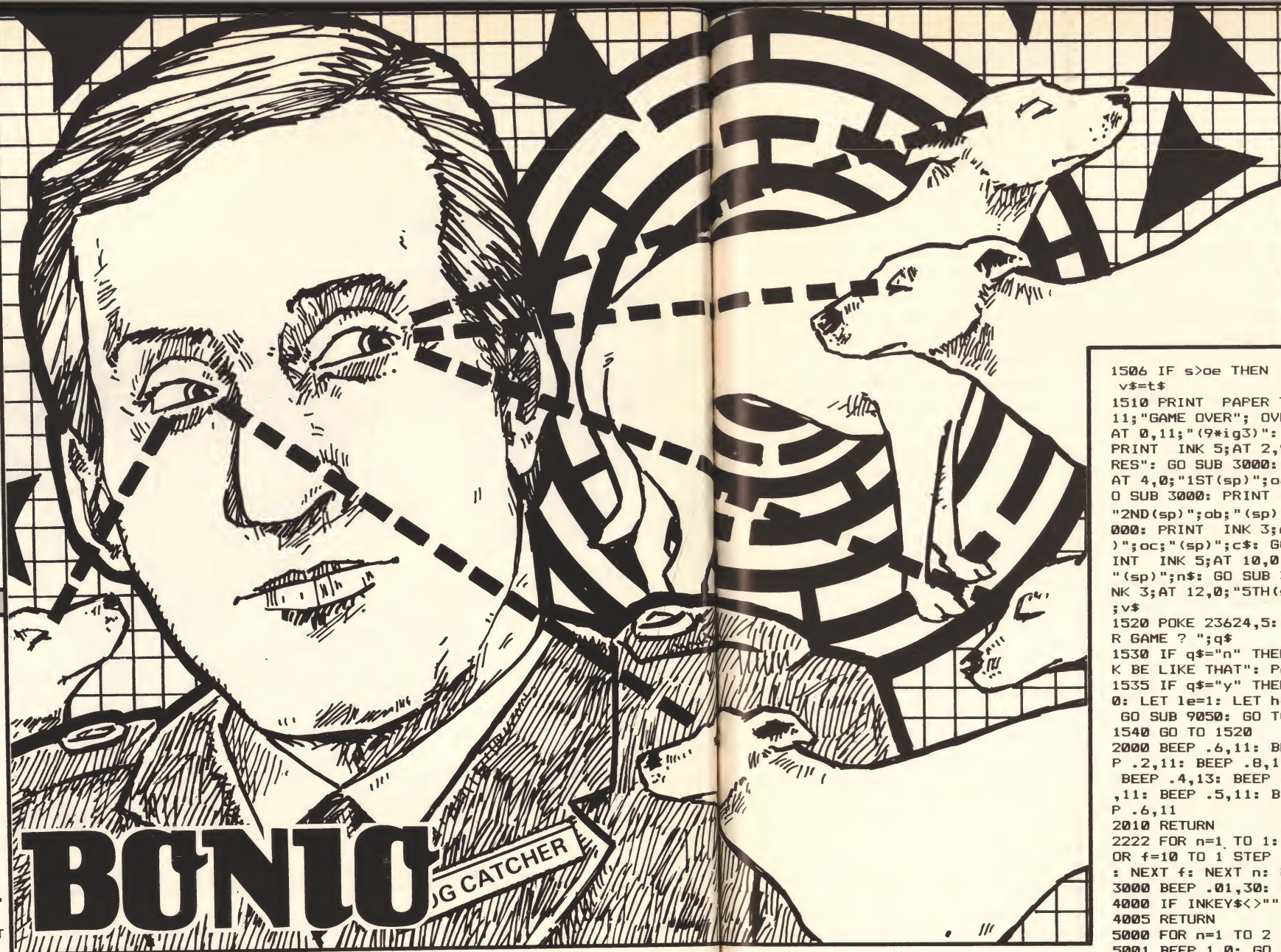
Bonio was written for the 48K Spectrum, and when you have completed one maze a new, more difficult level begins.

The program uses our special abbreviations for graphics characters, so please read the instructions on the first page of Program Printout before typing in the listing.

```

1 LET u=11: LET y=1
1: GO SUB 7000: LET le=1: LET h=
0: POKE 23562,1: LET t=999: GO S
UB 9000
2 LET a$="E"
3 PAPER 0: BORDER 0: INK 7: C
LS: PRINT PAPER 2: INK 7:AT 2,
5:"HIT ANY KEY TO START": OVER 1
: FLASH 1:AT 2,5:"(20*ig3)": GO
TO 5000
4 CLS: PRINT PAPER 2: INK 7
:"SCORE":AT 0,6:"00000":AT 0,14:
"TIME":AT 0,25:"LIVES"
5 LET s=0: LET l=3
7 LET i=11: LET k=27: LET g=1
9: DIM e$(22,32): GO SUB 8000
8 PRINT PAPER 0: INK 7:AT 10
,29:"C":AT 20,3:"C":AT 20,18:"C"
:AT 20,23:"C":AT 18,28:"C":AT 7,
7:"C":AT 3,26:"C":AT 7,15:"C":AT
14,17:"C":AT 20,11:"C"
10 LET a=3
15 PRINT PAPER 2: INK 7:AT 12
,0:"LEVEL":AT 12,6:le
20 LET b=4
25 IF s<10000 THEN PRINT PAP
ER 2: INK 7:AT 0,6:"0"
30 GO SUB 6000
31 PRINT PAPER 2: INK 7:AT 0,
19:t: PAPER 0:"(sp)": PAPER 2:AT
0,31:l
32 IF s<1000 THEN PRINT PAPE
R 2: INK 7:AT 0,8:s
33 IF s>1000 AND s<10000 THEN
PRINT PAPER 2: INK 7:AT 0,7:s
34 IF s>10000 THEN PRINT PAP
ER 2: INK 7:AT 0,6:s
39 PRINT PAPER 0: INK 7:AT a,
b:a$
40 IF INKEY$="2" AND e$(a-1,b+
1)<>"E" AND e$(a-2,b+1)<>"E" THE
N LET a=a-1: PRINT PAPER 0:AT
a+1,b:"(sp)"
50 IF INKEY$="w" AND e$(a-1,b+
1)<>"E" AND e$(a,b+1)<>"E" THEN
LET a=a+1: PRINT PAPER 0:AT a-
1,b:"(sp)"
60 IF INKEY$="c" AND e$(a-1,b)
<>"E" THEN LET b=b-1: LET a$="E"
: PRINT PAPER 0: INK 7:AT a,b+
1:"(sp)"
70 IF INKEY$="p" AND e$(a-1,b+
2)<>"E" THEN LET b=b+1: LET a$=
"E": PRINT PAPER 0: INK 7:AT a,
b-1:"(sp)"
80 IF INKEY$="h" THEN BORDER
6: BORDER 2: BORDER 1: GO TO 80
81 BORDER 0
90 IF INKEY$="a" THEN GO TO 1
500

```



```

196 IF le>=3 THEN PRINT INK 4
:AT 11,u-1:"D(sp)"
197 IF le>=2 THEN PRINT AT y-1
,0:"(sp)"
198 IF le>=2 THEN PRINT INK 4
:AT y,8:"D"
200 PRINT INK 4:AT 14,k-1:"(sp
)D"
201 PRINT INK 4:AT 11,g-1:"(sp
)D"
202 PRINT AT i-1,15:"(sp)"
203 PRINT INK 4:AT i,15:"D"
204 PRINT PAPER 0: INK 1:AT 11
,11:"E"
205 PRINT PAPER 0: INK 1:AT 14
,24:"E"
206 PRINT PAPER 0: INK 1:AT 11
,18:"E"
207 PRINT PAPER 0: INK 1:AT 10
,15:"E"
208 PRINT PAPER 0: INK 1:AT 10
,8:"E"
300 LET k=k+1: IF k=31 THEN PR
INT AT 14,k-1:"(sp)": LET k=25
301 LET g=g+1: IF g=25 THEN PR
INT AT 11,g-1:"(sp)": LET g=19
302 LET i=i+1: IF i=15 THEN PR
INT AT i-1,15:"(sp)": LET i=11
303 LET y=y+1: IF y=21 THEN PR
INT AT y-1,8:"(sp)": LET y=11
304 LET u=u-1: IF u=8 THEN PRI
NT AT 11,u:"(sp)": LET u=11
340 IF le>=3 AND a=11 AND b=u-1 THEN
GO SUB 500
345 IF le>=2 AND a=y AND b=8 TH
EN GO SUB 500
350 IF a=14 AND b=k-1 THEN GO

```

```

SUB 500
355 IF a=i-1 AND b=15 OR a=i AN
D b=15 THEN GO SUB 500
360 IF a=11 AND b=g-1 OR a=11 A
ND b=g THEN GO SUB 500
400 IF h=10 THEN GO TO 1000
410 IF l=0 THEN GO TO 1500
420 IF t<0 THEN LET l=1-1: GO
SUB 2000: LET t=999
430 LET t=t-1: GO TO 30
500 LET l=1-1: PRINT AT 1,1:"D"
: GO SUB 2000: LET a=3: LET b=4:
RETURN
1000 FOR n=1 TO 50: OUT 254,n: B
EEP .01,n: NEXT n: BORDER 0: CLS
1005 PRINT INK 6:AT 0,9:"WELL D
ONE !!!"
1007 BEEP .01,30
1010 PRINT INK 5:AT 5,9:"SCORE
=":AT 5,17:s
1011 BEEP .01,30
1020 PRINT INK 5:AT 7,9:"ADD"
1021 BEEP .01,30
1030 PRINT INK 5:AT 9,9:"TIME(2
*sp)":AT 9,17:t
1031 BEEP .01,30
1040 PRINT INK 5:AT 11,9:"ADD"
1041 BEEP .01,30
1045 LET b=le*1000
1050 PRINT INK 5:AT 13,9:"BONUS
=":AT 13,17:bo
1051 BEEP .01,30
1060 PRINT INK 6:AT 18,0:"LIVES
=":1:AT 19,0:1:"*500=":1*500
1061 BEEP .01,30
1070 LET s=s+t+1*500+bo
1080 LET le=le+1

```

```

1090 LET h=0
1095 GO SUB 9050
1100 LET t=1099-(le*100): FOR n=
1 TO 500: NEXT n: CLS
1111 PRINT PAPER 2: INK 7:"SCOR
E":AT 0,14:"TIME":AT 0,25:"LIVES
"
1120 GO TO 6
1500 POKE 23609,255: FOR n=1 TO
200: NEXT n: CLS: PRINT PAPER
7: INK 2:AT 0,11:"GAME OVER": OV
ER 1: FLASH 1:AT 0,11:"(9*ig3)":
GO SUB 3000: IF s>oe THEN PRIN
T INK 5:AT 2,11:
"WELL DONE": PRINT: GO SUB 3000
: PRINT INK 6:"YOU HAVE GOT ONE
OF TODAY": GO SUB 3000: PRINT
INK 6:"HIGHEST SCORES": GO SUB
3000: POKE 23624,5: GO SUB 3000:
INPUT "PLEASE INPUT YOUR NAME "
:t$
1501 CLS
1502 IF s>oa THEN LET oe=od: LE
T od=oc: LET oc=ob: LET ob=oa: L
ET v$=n$: LET n$=c$: LET c$=x$:
LET x$=z$: LET oa=s: LET z$=t$:
GO TO 1510
1503 IF s>ob THEN LET oe=od: LE
T od=oc: LET oc=ob: LET v$=n$: L
ET n$=c$: LET c$=x$: LET ob=s: L
ET x$=t$: GO TO 1510
1504 IF s>oc THEN LET oe=od: LE
T od=oc: LET oc=ob: LET v$=n$: L
ET n$=c$: LET oc=s: LET c$=t$: G
O TO 1510
1505 IF s>od THEN LET oe=od: LE
T od=oc: LET oc=ob: LET v$=n$: L
ET od=s: LET n$=t$: GO TO 1510

```

```

1506 IF s>oe THEN LET oe=s: LET
v$=t$
1510 PRINT PAPER 7: INK 2:AT 0,
11:"GAME OVER": OVER 1: FLASH 1:
AT 0,11:"(9*ig3)": GO SUB 3000:
PRINT INK 5:AT 2,9:"HIGHEST SCO
RES": GO SUB 3000: PRINT INK 3:
AT 4,0:"1ST(sp)":oa:"(sp)":z$: G
O SUB 3000: PRINT INK 5:AT 6,0:
"2ND(sp)":ob:"(sp)":x$: GO SUB 3
000: PRINT INK 3:AT 8,0:"3RD(sp
)":oc:"(sp)":c$: GO SUB 3000: PR
INT INK 5:AT 10,0:"4TH(sp)":od:
"(sp)":n$: GO SUB 3000: PRINT I
NK 3:AT 12,0:"5TH(sp)":oe:"(sp)"
:v$
1520 POKE 23624,5: INPUT "ANOTHE
R GAME ?":q$
1530 IF q$="n" THEN PRINT #1:"O
K BE LIKE THAT": PAUSE 500: NEW
1535 IF q$="y" THEN POKE 23609,
0: LET le=1: LET h=0: LET t=999:
GO SUB 9050: GO TO 2
1540 GO TO 1520
2000 BEEP .6,11: BEEP .6,11: BEE
P .2,11: BEEP .8,11: BEEP .5,14:
BEEP .4,13: BEEP .4,13: BEEP .4
,11: BEEP .5,11: BEEP .3,11: BEE
P .6,11
2010 RETURN
2222 FOR n=1 TO 1: BEEP .01,n: F
OR f=10 TO 1 STEP -1: BEEP .01,f
: NEXT f: NEXT n: RETURN
3000 BEEP .01,30: RETURN
4000 IF INKEY$<>" " THEN GO TO 4
0005 RETURN
5000 FOR n=1 TO 2
5001 BEEP 1,0: GO SUB 4000
5002 BEEP 1,2: GO SUB 4000
5003 BEEP .5,3: GO SUB 4000
5004 BEEP .5,2: GO SUB 4000
5005 BEEP 1,0: GO SUB 4000
5006 NEXT n
5007 FOR n=1 TO 2
5008 BEEP 1,3: GO SUB 4000
5009 BEEP 1,5: GO SUB 4000
5010 BEEP 2,7: GO SUB 4000
5011 NEXT n
5012 FOR n=1 TO 2
5013 BEEP .75,7: GO SUB 4000
5014 BEEP .25,8: GO SUB 4000
5015 BEEP .5,7: GO SUB 4000
5016 BEEP .5,5: GO SUB 4000
5017 BEEP .5,3: GO SUB 4000
5018 BEEP .5,2: GO SUB 4000
5019 BEEP 1,0: GO SUB 4000
5020 NEXT n
5021 FOR n=1 TO 2
5022 BEEP 1,0: GO SUB 4000
5023 BEEP 1,-5: GO SUB 4000
5024 BEEP 2,0: GO SUB 4000
5025 NEXT n
5030 PAUSE 0: GO TO 4
6010 IF aa=1 AND a=10 AND b=29 T
HEN LET s=s+150: LET aa=2: LET h
=h+1: GO SUB 2222
6015 IF bb=1 AND a=20 AND b=3 TH
EN LET s=s+150: LET bb=2: LET h
=h+1: GO SUB 2222
6020 IF cc=1 AND a=20 AND b=18 T
HEN LET s=s+150: LET cc=2: LET
h=h+1: GO SUB 2222
6025 IF dd=1 AND a=20 AND b=23 T
HEN LET s=s+150: LET dd=2: LET
h=h+1: GO SUB 2222
6030 IF ee=1 AND a=18 AND b=28 T

```

```

HEN LET s=s+150: LET ee=2: LET
h=h+1: GO SUB 2222
6035 IF ff=1 AND a=7 AND b=7 THE
N LET s=s+150: LET ff=2: LET h=
h+1: GO SUB 2222
6040 IF gg=1 AND a=3 AND b=26 TH
EN LET s=s+150: LET gg=2: LET h
=h+1: GO SUB 2222
6045 IF hh=1 AND a=7 AND b=15 TH
EN LET s=s+150: LET hh=2: LET h
=h+1: GO SUB 2222
6055 IF jj=1 AND a=14 AND b=17 T
HEN LET s=s+150: LET jj=2: LET
h=h+1: GO SUB 2222
6060 IF kk=1 AND a=20 AND b=11 T
HEN LET s=s+150: LET kk=2: LET
h=h+1: GO SUB 2222
6070 RETURN
7000 LET oa=0: LET ob=0: LET oc=
0: LET od=0: LET oe=0: LET z$="Z
X SPECTRUM": LET x$=z$: LET c$=z
$: LET n$=z$: LET v$=z$
7010 RETURN
8000 LET e$(1)="(32*E)"
8010 LET e$(2)="(4*E,sp,5*E,7*sp
,7*E,4*sp,4*E)"
8020 LET e$(3)="(E,10*sp,5*E,6*sp
p,2*E,sp,2*E,4*sp,E)"
8030 LET e$(4)="(E,sp,8*E,sp,10*
E,sp,2*E,sp,5*E,sp,E)"
8040 LET e$(5)="(E,sp,8*E,sp,10*
E,sp,2*E,sp,5*E,sp,E)"
8050 LET e$(6)="(E,sp,3*E,3*sp,2
*E,sp,2*E,9*sp,2*E,7*sp,E)"
8060 LET e$(7)="(E,sp,3*E,sp,4*E
,sp,2*E,sp,5*E,sp,4*E,sp,5*E,sp
,E)"
8070 LET e$(8)="(E,sp,3*E,sp,4*E
,sp,2*E,sp,5*E,sp,4*E,sp,5*E,sp
,E)"
8080 LET e$(9)="(E,5*sp,4*E,sp,2
*E,sp,5*E,sp,4*E,sp,2*E,4*sp,E)"
8090 LET e$(10)="8*E,3*sp,2*E,3*
sp,3*E,6*sp,2*E,sp,4*E)"
8100 LET e$(11)="(8*E,sp,6*E,sp,
6*E,sp,4*E,sp,4*E)"
8110 LET e$(12)="(8*E,sp,6*E,sp,
6*E,sp,4*E,sp,4*E)"
8120 LET e$(13)="(E,8*sp,2*E,8*sp
p,3*E,sp,2*E,6*sp,E)"
8130 LET e$(14)="(E,sp,6*E,sp,2*
E,sp,6*E,5*sp,2*E,sp,4*E,sp,E)"
8140 LET e$(15)="(E,sp,6*E,sp,2*
E,sp,6*E,sp,6*E,sp,4*E,sp,E)"
8150 LET e$(16)="(E,sp,6*E,sp,2*
E,3*sp,4*E,sp,6*E,sp,4*E,sp,E)"
8160 LET e$(17)="(E,sp,6*E,sp,4*
E,sp,2*E,3*sp,2*E,5*sp,2*E,sp,E
,sp,E)"
8170 LET e$(18)="(E,sp,6*E,sp,4*
E,sp,2*E,sp,4*E,sp,6*E,sp,E,sp,
E)"
8180 LET e$(19)="(E,3*sp,4*E,6*sp
p,2*E,3*sp,2*E,3*sp,4*E,3*sp,E)"
8190 LET e$(20)="(33*E)"
8200 PRINT: FOR n=1 TO 20: PRIN
T PAPER 0: INK 1:e$(n): BEEP .0
1,n+1: NEXT n: PRINT PAPER 1: I
NK 2:AT 2,4:"E": RETURN
9000 FOR n=USR "a" TO USR "g"+7:
READ q: POKE n,q: NEXT n
9010 DATA 0,0,48,241,255,63,35,1
02
9030 DATA 0,0,12,143,255,252,196
,102,0,0,195,231,126,231,195,0,5
6,56,56,16,124,16,40,108
9040 DATA 255,255,255,231,231,25
5,255,255
9045 DATA 0,0,0,0,24,60,126,255,
8,28,8,28,62,62,62
9050 LET aa=1: LET bb=1: LET cc=
1: LET dd=1: LET ee=1: LET ff=1:
LET gg=1: LET hh=1: LET ii=1: L
ET jj=1: LET kk=1
9990 RETURN
9998 STOP
9999 SAVE "Bonio" LINE 1

```


SIXTY-FOUR COLUMN text on a Spectrum? Impossible? Not with **Word Processor** by Mark Lawrence of Basildon in Essex. The program is written entirely in machine code, and offers the following facilities.

With CAPS SHIFT:

- 0—Delete character
- 1—Clear all text from memory
- 2—Switch on/off CAPS LOCK
- 3—Move back one page
- 4—Move forward one page
- 5—Move cursor left
- 6—Move cursor down
- 7—Move cursor up
- 8—Move cursor right

In Extended Mode (CAPS SHIFT and SYMBOL SHIFT):

B—Change border, paper and ink colours. You will be asked for a three digit number. The first is the border colour, the second the paper, and the third ink. You may not use digits greater than seven.

F—Find the first occurrence of a string. You must input the required string.

G—Goto page. The 25 pages are labelled A–Y.

L—Load text. You may load text either from tape or microdrive. The text is loaded from page A onwards, so you must move them elsewhere if you want to save text they contain already.

M—Move text from one page to another. The original page is not deleted with this command. You will be asked to give the letters for both pages.

P—Print page. This will print the current page number and wait for you to press a key.

R—Send to RS232. This will send the current page to the RS 232 interface on Interface One.

S—Save text. You can save to tape or microdrive. You will be asked for starting and finishing points of the text and given the opportunity to VERIFY.

Z—Copy current page to ZX Printer.

To enter the program, first type in listing 1 and save it with SAVE "w/p" LINE 10. Now delete the program from the computer and enter listing 2. If you want to save Listing 2, do so on a separate tape. RUN the program and enter the machine code in listing 3 as requested. The last number in each row is a checksum and if there is a mistake you will be told to type in that row again. When you have entered all the code, enter 999 and the computer will SAVE the code.

Listing 1

```
10 INK 7: PAPER 1: BORDER 0: C
LEAR 24999: LOAD ""CODE
20 RANDOMIZE USR 25000
```

Listing 2

```
10 REM loader
20 LET start=25000
25 LET tot=0
30 FOR x=1 TO 7
40 INPUT "Enter code ";a
45 IF a=999 THEN GO TO 600
50 POKE start,a
60 PRINT start,a
70 LET start=start+1
80 LET tot=tot+a
90 NEXT x
100 INPUT "Enter checksum ";che
ck
110 IF tot<>check THEN GO TO 5
00
120 GO TO 25
500 PRINT "You have made an err
or." "Please retype the line you
have" "just entered"
510 LET start=start-7
520 GO TO 25
600 SAVE "word code"CODE 25000,
2135
```

Listing 3

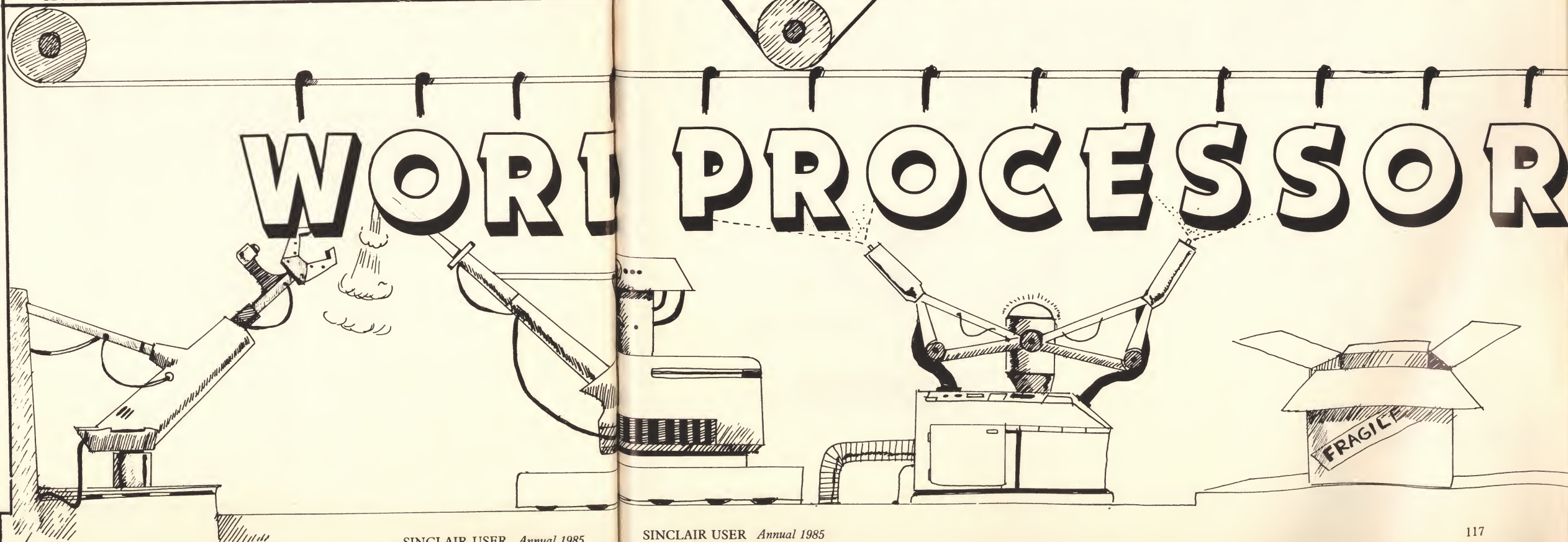
```
33 0 64 34 124 104 175 534
33 113 104 119 35 119 33 556
8 92 119 205 157 98 205 884
32 99 205 3 99 33 8 479
92 126 254 14 40 54 214 794
32 218 59 99 254 96 48 806
236 54 0 205 90 100 205 890
241 102 205 123 101 42 124 938
104 205 209 102 30 1 205 856
129 100 58 113 104 181 254 939
0 32 4 124 198 7 103 468
34 124 104 205 3 99 205 774
32 99 24 191 229 205 123 903
101 225 205 32 99 175 119 956
126 254 0 40 251 205 123 999
101 203 175 254 0 200 254 1187
83 204 254 100 254 76 202 1173
21 101 254 10 202 203 98 889
254 82 204 223 98 254 66 1181
204 88 98 254 80 204 193 1121
98 254 71 204 180 98 254 1159
77 204 142 98 254 70 204 1049
43 103 254 90 32 6 243 771
6 192 205 175 14 205 32 829
99 175 33 8 92 119 24 550
166 62 7 205 235 100 205 980
138 103 17 115 104 205 211 893
98 48 36 205 151 34 19 591
205 211 98 48 27 203 39 831
92 126 230 7 129 119 19 722
205 211 98 48 6 79 126 773
230 56 129 119 24 15 205 778
178 100 229 9 1 0 6 523
167 237 66 235 225 237 176 1343
58 114 104 205 148 102 33 764
0 88 17 1 88 1 0 195
3 58 141 92 119 237 176 826
175 201 62 1 205 235 100 979
205 133 102 50 114 104 24 732
220 58 114 104 198 193 50 937
48 104 62 6 205 235 100 760
205 111 101 24 202 26 214 883
48 56 5 254 8 48 1 420
201 63 201 58 114 104 205 946
217 100 1 0 6 197 229 750
121 230 63 32 6 62 13 527
207 30 225 229 126 198 32 1047
207 30 225 193 35 11 120 821
177 32 230 201 58 125 104 927
```

```
254 88 56 7 33 255 80 773
62 1 24 8 254 64 48 461
10 33 0 64 175 34 124 440
104 50 113 104 201 245 229 1046
14 240 58 113 104 254 0 783
40 2 14 15 6 8 42 127
124 104 126 169 119 36 16 694
250 225 241 201 126 254 12 1309
32 39 205 32 99 42 124 573
104 30 255 205 129 100 58 881
113 104 254 1 32 9 125 638
254 255 32 4 124 214 7 890
103 34 124 104 175 205 241 986
102 205 209 102 24 35 254 931
13 32 33 205 32 99 42 456
124 104 125 230 224 111 1 919
32 0 254 224 32 3 1 546
32 7 9 34 124 104 205 515
3 99 175 50 113 104 24 568
29 254 8 32 27 42 124 516
104 125 230 31 254 0 32 776
7 58 113 104 254 0 40 576
29 205 32 99 30 255 205 855
129 100 24 59 254 9 32 607
27 42 124 104 125 230 31 683
254 31 32 7 58 113 104 599
254 1 40 79 205 32 99 710
30 1 205 129 100 24 28 517
254 10 32 26 205 32 99 658
42 124 104 1 32 0 125 428
214 224 56 3 1 32 7 537
9 34 124 104 205 3 99 578
24 101 254 11 32 28 205 655
32 99 42 124 104 1 32 434
0 125 214 32 48 3 1 423
32 7 167 237 66 34 124 667
104 205 3 99 24 69 254 758
6 32 5 205 221 16 24 509
63 254 4 32 22 58 114 547
104 254 0 40 52 61 50 561
114 104 205 148 102 33 0 706
```

```
64 34 124 104 24 34 254 638
5 32 10 58 114 104 254 577
24 40 26 60 24 228 254 656
7 32 26 33 254 105 17 474
255 105 1 0 150 54 0 565
237 176 195 168 97 205 32 1110
99 205 123 101 175 50 8 761
92 195 191 97 245 205 97 1122
100 241 119 201 42 124 104 931
124 214 64 203 63 203 63 934
203 63 103 41 237 91 113 851
104 122 22 0 25 235 205 713
227 100 25 17 254 105 25 753
201 1 113 104 10 131 230 790
1 2 131 254 0 32 1 421
43 254 1 32 1 35 34 400
124 104 201 221 33 176 92 951
17 2 0 175 55 205 86 540
5 221 33 254 105 237 91 946
176 92 62 255 55 195 86 921
5 6 2 197 62 2 144 418
205 235 100 205 133 102 193 1173
120 254 2 32 4 123 50 585
115 104 16 234 58 115 104 746
79 123 145 60 205 227 100 939
229 193 58 115 104 205 227 1131
100 213 17 254 105 25 209 923
201 135 103 135 132 103 46 855
0 201 245 205 107 13 241 1012
17 203 103 195 10 12 62 602
2 205 235 100 195 111 101 949
205 246 100 254 84 32 5 926
205 41 101 24 8 254 77 710
205 51 102 205 83 102 195 943
157 98 205 246 100 254 84 1144
32 5 205 152 100 24 5 523
254 77 204 212 101 195 168 1211
97 205 178 100 197 229 237 1243
67 176 92 17 2 0 221 575
33 176 92 175 205 194 4 879
221 225 209 213 221 229 62 1380
```

```
255 205 194 4 205 139 101 1103
56 58 17 2 0 221 33 387
176 92 175 205 86 5 48 787
10 221 225 209 62 255 167 1149
205 86 5 216 205 107 13 837
17 145 19 62 27 205 10 485
12 33 8 92 54 0 126 325
254 0 40 251 203 175 245 1168
33 100 0 17 3 0 205 358
181 3 241 201 221 225 209 1281
201 62 8 205 235 100 205 1016
111 101 254 89 201 207 49 1012
193 217 229 217 197 62 1 1116
50 214 92 205 183 101 34 879
218 92 33 115 104 34 220 816
92 207 34 221 229 225 34 1042
81 92 201 62 4 205 235 880
100 205 138 103 17 115 104 782
33 0 0 26 254 13 40 366
4 19 35 24 247 125 254 708
0 40 228 201 205 150 101 925
221 229 62 3 205 235 100 1055
225 34 81 92 1 0 0 433
33 254 105 229 205 0 102 928
225 254 255 40 4 119 35 932
24 243 207 35 62 2 205 778
1 22 217 225 217 201 237 1120
67 115 104 205 230 21 221 963
229 245 121 254 0 32 25 906
62 1 205 1 22 62 22 375
215 62 1 215 62 12 215 782
237 75 115 104 120 205 40 896
45 205 227 45 241 225 34 1022
81 92 237 75 115 104 3 707
201 205 178 100 229 197 205 1315
150 101 209 193 225 213 34 1125
176 92 197 229 126 215 225 1260
193 35 11 120 177 32 244 812
62 255 215 24 162 205 139 1062
101 216 217 229 217 207 34 1221
```

Listing continues on next page



continued from page 117

```
221 229 62 9 205 235 100 1061
225 34 81 92 1 0 0 433
42 176 92 229 205 0 102 846
225 254 255 40 9 190 32 1005
3 35 24 241 205 100 101 709
207 35 217 225 217 201 229 1331
205 111 101 214 65 56 249 1001
254 25 48 245 95 225 201 1093
205 217 100 235 33 0 64 854
175 50 113 104 1 0 6 449
229 213 197 235 126 205 241 1446
102 193 209 225 205 209 102 1245
19 58 113 104 254 1 32 581
10 35 125 254 0 32 4 460
124 198 7 103 58 113 104 707
60 230 1 50 113 104 11 569
120 177 32 210 201 197 213 1150
229 17 115 104 58 113 104 740
14 15 254 1 32 2 14 332
240 6 8 235 26 161 134 810
18 35 20 16 248 225 209 771
193 201 229 33 113 104 78 951
111 38 0 41 41 17 126 374
104 25 17 115 104 6 4 375
126 230 240 203 65 40 8 912
203 63 203 63 203 63 203 1001
63 18 19 126 230 15 203 674
65 32 8 203 39 203 39 589
203 39 203 39 18 35 19 556
16 219 225 201 62 5 205 933
235 100 205 138 103 33 254 1068
105 205 118 103 254 13 40 838
7 35 124 181 32 244 24 647
3 205 74 103 195 157 98 835
1 254 105 175 237 66 1 839
0 6 60 167 237 66 48 584
250 61 50 114 104 9 125 713
230 1 50 113 104 203 60 761
203 29 124 203 39 203 39 840
203 39 198 64 103 34 124 765
104 201 229 17 115 104 26 796
254 13 40 9 214 32 190 752
32 4 35 19 24 242 225 581
201 17 115 104 33 18 92 570
```

```
54 0 126 254 32 48 28 542
254 13 40 46 254 12 32 651
243 33 115 104 167 237 82 981
40 230 62 8 215 62 32 649
215 62 8 215 27 24 15 566
254 128 48 219 33 123 104 909
167 237 82 40 206 18 19 769
215 213 205 123 101 209 24 1090
196 18 201 128 70 114 111 838
109 32 119 104 105 99 104 672
32 80 97 103 101 32 191 636
84 111 32 119 104 105 99 654
104 32 112 97 103 101 32 581
191 77 105 99 114 111 100 797
114 105 118 101 32 111 114 695
32 116 97 112 101 32 63 553
32 40 109 47 116 169 76 589
111 97 100 105 110 103 46 672
174 70 105 108 101 32 110 700
97 109 101 32 63 160 83 645
101 97 114 99 104 32 83 630
116 114 105 110 103 32 63 643
160 80 97 103 101 61 160 762
69 110 116 101 114 32 97 639
116 116 114 105 98 117 116 782
101 115 32 40 66 44 80 478
44 73 41 160 86 101 114 619
105 102 121 32 63 32 40 495
89 47 78 169 86 101 114 684
105 102 121 105 110 231 127 901
32 77 46 80 46 76 32 389
83 83 49 54 32 52 74 427
194 0 0 0 0 0 0 194
0 0 0 0 0 0 0 64 64
0 0 0 0 4 68 64 136
64 5 160 0 0 0 174 403
174 160 4 104 66 196 0 704
222 123 0 0 74 74 96 589
2 64 0 0 4 136 136 342
64 4 34 34 64 0 164 364
228 160 0 4 228 0 0 620
0 68 128 0 0 224 0 420
0 0 0 128 2 36 72 238
128 4 170 234 64 4 196 800
68 224 4 162 72 224 14 768
```

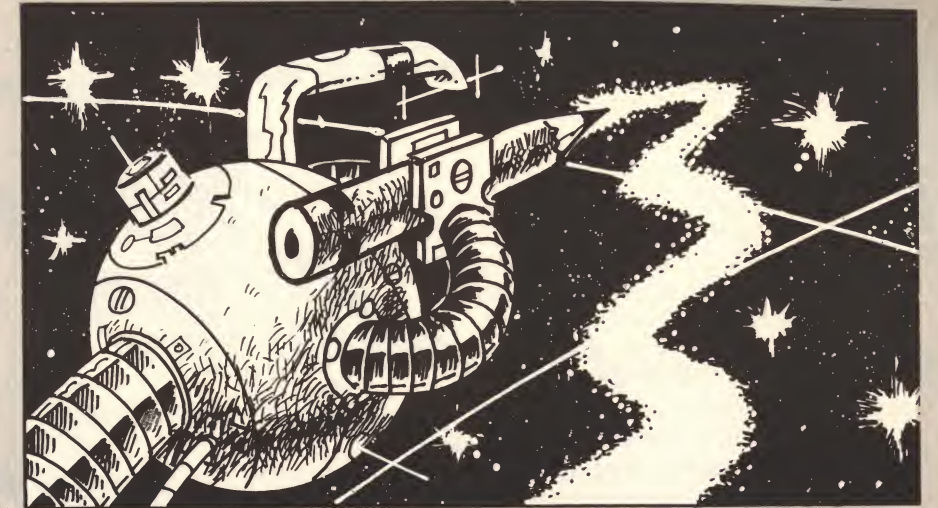
```
36 42 64 2 106 175 32 457
14 140 42 64 6 140 170 576
64 14 34 68 64 4 164 412
170 64 6 170 98 32 0 540
64 64 0 0 64 68 128 388
0 36 132 32 0 14 14 228
0 0 132 36 128 14 36 346
64 64 0 78 232 96 4 538
170 234 160 12 172 170 192 1110
4 168 138 64 12 170 170 726
192 14 140 136 224 14 140 860
136 128 4 168 234 96 10 776
170 234 160 14 68 68 224 938
14 68 68 128 10 172 202 662
160 8 136 136 224 10 234 908
170 160 10 174 170 160 4 848
170 170 64 12 170 200 128 914
4 170 174 96 12 170 202 828
160 6 132 34 192 14 68 606
68 64 10 170 170 64 10 556
170 164 64 10 170 174 160 912
10 164 74 160 10 164 68 650
64 14 36 72 224 12 136 558
136 192 8 132 66 32 12 578
68 68 192 4 228 68 64 692
0 0 0 224 6 136 200 566
224 0 66 106 96 0 136 628
202 192 0 74 138 64 0 670
34 106 96 0 74 200 96 606
0 104 200 128 0 106 98 636
192 0 136 202 160 0 64 754
68 64 0 34 42 64 0 272
170 202 160 0 136 136 96 908
0 174 170 160 0 74 170 748
160 0 74 170 64 0 202 670
200 128 0 106 98 50 0 582
104 136 128 0 104 66 192 730
0 140 138 64 0 170 170 682
64 0 170 164 64 0 170 632
174 160 0 170 74 160 0 738
170 98 164 0 226 72 224 954
2 72 68 32 4 68 68 314
64 8 66 68 128 5 160 499
0 0 6 159 223 96 0 484
```

SPACE-AGE CHARACTERS

IF YOU ARE writing a space game and do not want the tedium of altering the whole character set, then try **Space-Age Characters** by David Nunn of Sudbury in Suffolk.

The program runs on the 48K Spectrum, but could easily be adapted to 16K by altering the value in lines 20, 70, 190, 470, 750, and 820 to values suitable to the smaller machine. We suggest using 31,999 as the initial value; the rest should be altered proportionally.

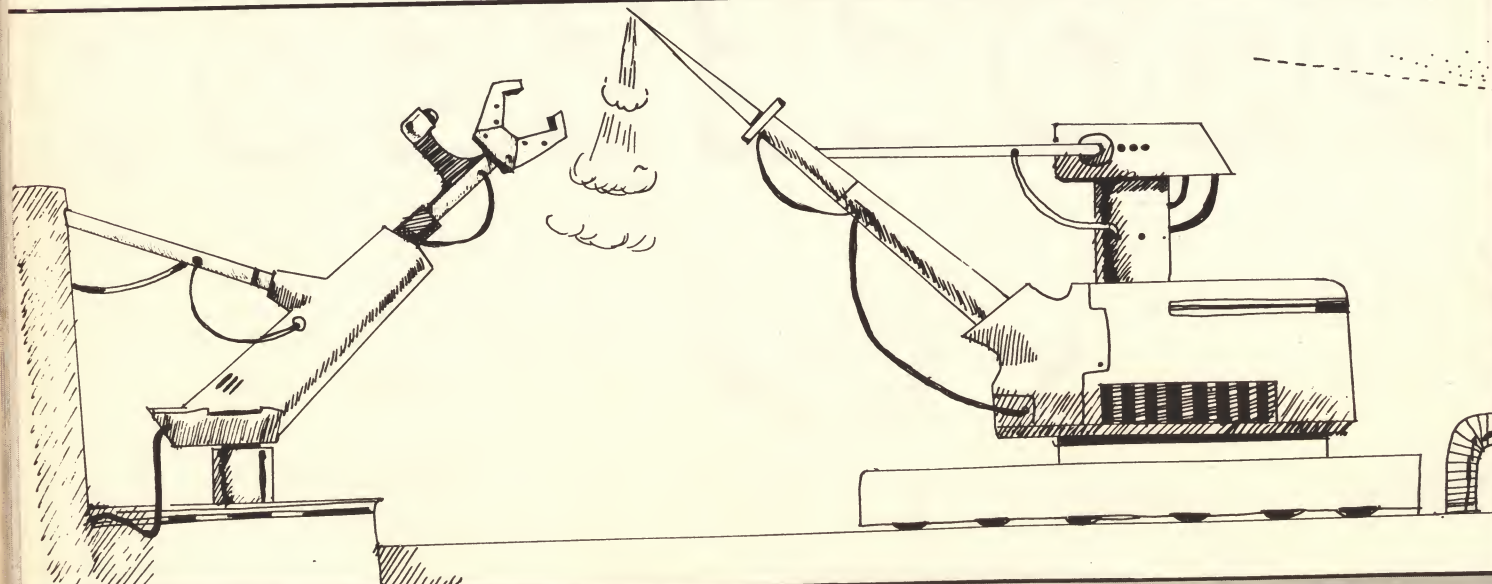
Once the program has been RUN, the computer will ask you if you want to SAVE the generated character set, and will then tell you how to use the SAVED data in a program.



```
20 CLEAR 63999
30 INK 7: PAPER 0: BORDER 0: C
LS
40 PRINT AT 11,9; FLASH 1;" P1
ease Wait "
50 FOR f=0 TO 767: POKE 64000+
f,PEEK (15616+f): NEXT f
60 REM 0123456789
70 FOR f=64128 TO 64207: READ
a: POKE f,a: NEXT f
80 DATA 0,126,102,106,106,114,
126,0
90 DATA 0,24,120,24,24,24,126,
0
100 DATA 0,126,2,2,126,96,126,0
110 DATA 0,126,6,62,6,6,126,0
120 DATA 0,96,96,100,126,4,4,0
130 DATA 0,126,96,96,126,2,126,
0
140 DATA 0,126,96,96,126,98,126,
0
150 DATA 0,126,6,6,6,6,6,0
160 DATA 0,126,98,126,98,98,126,
0
170 DATA 0,126,98,98,126,2,2,0
180 REM ABCDEFGHIJKLM
NOPQRSTUVWXYZ
190 FOR f=64264 TO 64471: READ
a: POKE f,a: NEXT f
200 DATA 0,126,98,98,126,98,98,
0
210 DATA 0,126,98,124,98,98,126,
0
220 DATA 0,126,96,96,96,96,126,
0
230 DATA 0,124,98,98,98,98,124,
0
240 DATA 0,126,96,124,96,96,126,
0
250 DATA 0,126,96,124,96,96,96,
0
260 DATA 0,126,96,96,102,98,126,
0
270 DATA 0,98,98,126,98,98,98,0
280 DATA 0,126,24,24,24,24,126,
0
290 DATA 0,126,24,24,24,24,120,
0
300 DATA 0,100,104,112,104,100,
98,0
310 DATA 0,96,96,96,96,96,126,0
320 DATA 0,126,106,106,106,106,
106,0
330 DATA 0,98,114,122,106,110,1
02,0
340 DATA 0,126,98,98,98,98,126,
0
350 DATA 0,126,98,98,126,96,96,
0
```

```
360 DATA 0,126,98,98,106,102,12
6,0
370 DATA 0,126,98,98,126,104,10
2,0
380 DATA 0,126,96,126,2,2,126,0
390 DATA 0,126,24,24,24,24,24,0
400 DATA 0,98,98,98,98,98,126,0
410 DATA 0,98,98,98,98,52,24,0
420 DATA 0,106,106,106,106,106,
126,0
430 DATA 0,102,102,24,102,102,1
02,0
440 DATA 0,98,98,98,126,8,8,0
450 DATA 0,126,12,24,48,96,126,
0
460 REM abcdefghijklm
nopqrstuvwxyz
470 FOR f=64520 TO 64727: READ
a: POKE f,a: NEXT f
480 DATA 0,0,124,4,124,100,124,
0
490 DATA 0,96,96,124,100,100,12
4,0
500 DATA 0,0,124,96,96,96,124,0
510 DATA 0,4,4,124,100,100,124,
0
520 DATA 0,0,124,100,124,96,124,
0
530 DATA 0,28,48,56,48,48,48,0
540 DATA 0,0,124,100,100,124,4,
60
550 DATA 0,96,96,124,100,100,10
0,0
560 DATA 0,48,0,48,48,48,120,0
570 DATA 0,24,0,24,24,24,24,112
580 DATA 0,96,104,112,112,104,1
00,0
590 DATA 0,48,48,48,48,48,28,0
600 DATA 0,0,126,106,106,98,98,
0
610 DATA 0,0,124,100,100,100,10
0,0
620 DATA 0,0,124,100,100,100,12
4,0
630 DATA 0,0,124,100,100,124,96
,96
640 DATA 0,0,124,100,100,124,4,
6
650 DATA 0,0,28,48,48,48,48,0
660 DATA 0,0,124,96,124,4,124,0
670 DATA 0,48,120,48,48,48,28,0
680 DATA 0,0,100,100,100,100,12
4,0
690 DATA 0,0,100,100,100,56,16,
0
700 DATA 0,0,98,98,106,106,126,
0
710 DATA 0,0,108,108,16,108,108
,0
720 DATA 0,0,100,100,100,124,4,
60
730 DATA 0,0,124,12,56,96,124,0
740 REM ()?E!
750 FOR f=64064 TO 64079: READ
```

```
a: POKE f,a: NEXT f
760 DATA 0,28,24,24,24,24,28,0
770 DATA 0,56,24,24,24,24,56,0
780 FOR f=64248 TO 64255: READ
a: POKE f,a: NEXT f
790 DATA 0,124,4,60,48,0,48,0
800 FOR f=64512 TO 64519: READ
a: POKE f,a: NEXT f
810 DATA 0,62,48,120,48,48,126,
0
820 FOR f=64008 TO 64015: READ
a: POKE f,a: NEXT f
830 DATA 0,24,24,24,24,0,24,0
1000 POKE 23606,0: POKE 23607,24
9
1010 BORDER 1: PAPER 5: INK 1: C
LS
1020 PRINT AT 1,7; INK 2;"Your n
ew character";AT 3,6;"set has be
en created"
1030 PRINT AT 7,11; INVERSE 1;"0
123456789";AT 9,3;"ABCDEFGHIJKLM
NOPQRSTUVWXYZ";AT 11,3;"abcdegh
ijklmnopqrstuvwxyz"
1040 PRINT AT 15,1;"Press ""S""
when you are ready";AT 16,1;"to
save the new character set";AT 1
7,1;"on a spare tape."
1050 BEEP .3,30
1060 PAUSE 0
1070 IF INKEY$="s" OR INKEY$="S"
THEN SAVE "Characters"CODE 640
00,768: GO TO 1090
1080 GO TO 1060
1090 BORDER 2: INK 2: PAPER 6: C
LS
1100 PRINT AT 2,7; INK 1;"Your n
ew character";AT 4,7;"set has be
en saved"
1110 PRINT AT 6,1;"To use the ne
w character set";AT 7,1;"in non
machine-code games,";AT 8,1;"Wit
h ordinary characters,";AT 9,1;"
Use the following program:-"
1120 PRINT AT 11,3;"10 LOAD """"
CODE ";AT 12,3;"20 CLEAR 63999";
AT 13,3;"30 POKE 23606,0";AT 14,
3;"40 POKE 23607,249";AT 15,3;"5
0 LOAD """"
1130 PRINT AT 17,1;"Line 10 load
s the character";AT 18,1;"set yo
u have just saved."
1140 PRINT AT 20,1;"Line 50 load
s your game."
1150 PRINT#0;" Press""Z""for
a printer copy. Press""
S""to stop."
1160 BEEP .3,30
1170 PAUSE 0
1180 IF INKEY$="z" OR INKEY$="Z"
THEN COPY
1190 IF INKEY$="s" OR INKEY$="S"
THEN STOP
1200 GO TO 1170
```



Artificial Intelligence

ARTIFICIAL Intelligence is a version of a classic mainframe program, Terry Winograd's SHRDLU. Peter Cooke of Leicester has written this version for the 48K Spectrum. The program prints a number of objects on the screen, defined by their shape and colour. You then type in sentences instructing the computer to move the objects around. For example, when typing "Move the green triangle left" the computer would obey, but if you asked "Move the triangle left" the computer would say "I can see two triangles. Which do you mean?"

The importance of SHRDLU was that it enabled a computer to make intelligent deductions and act accordingly using spatial relationships as data within the limits of simple standard English. That has obvious implications for a computer's ability to recognise its environment, as well as for aspects of robot control.

The program uses our special abbreviations for graphics characters, so please read the instructions on the first page of the Program Printout section before you type in the program.

```

2 POKE 23658,0: REM caps off
3 DEF FN s$(n)="(sp)"+("small"
  AND n=1)+("big" AND n=2)
4 DEF FN c$(n)="(sp)"+("blue"
  AND n=1)+("red" AND n=2)+("magenta"
  AND n=3)+("green" AND n=4)+("cyan"
  AND n=5)+("yellow" AND n=6)
5 DEF FN n$(n)="(sp)"+("square"
  AND n=1)+("oblong" AND n=2)+("triangle"
  AND n=3)+("circle" AND n=4)
6 DEF FN a$(n)=FN s$(o(n,1))+FN c$(o(n,2))+FN n$(o(n,3))
7 DEF FN q$(n)="(sp)"+("any"
  AND n=0)+("one" AND n=1)+("two"
  AND n=2)+("three" AND n=3)+("four"
  AND n=4)+("five" AND n=5)+("six"
  AND n=6)+("seven" AND n=7)+("eight"
  AND n=8)+("nine" AND n=9)+("ten"
  AND n=10)
8 DEF FN h(x)=2+(1 AND o(x,1))
  +2+(1 AND o(x,3))
10 IF PEEK(USR "o"+5)>201 THEN
  GO SUB 9400: REM Check if m/
  code in place
20 PAPER 7: INK 0: BORDER 7: B
  RIGHT 0: INVERSE 0: FLASH 0: OVE
  R 0: CLS
30 PRINT AT 10,0;"(32*N)"
40 GO SUB 8000
50 PRINT AT 12,0;"Type help or
  some other command"
100 GO SUB 2000: LET v=USR USR
  "o": GO SUB 200*action: LET v=US

```

```

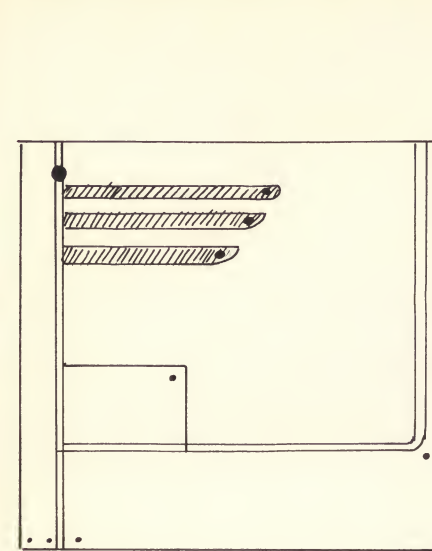
R USR "o"
110 GO TO 100
200 LET k=x$: GO SUB 7300: REM
  "list"
210 IF nomat=0 THEN GO SUB 7100:
  RETURN
220 LET z$="I can see ": GO SUB
  7400
230 FOR n=1 TO nosha: IF o(n,1)
  <>0 AND f(n) THEN LET z$=" a"+FN
  a$(n): GO SUB 7400: IF o(n,5)
  <>0 THEN LET z$="which is on top
  of the"+FN a$(o(n,5)): GO SUB 7
  400
240 NEXT n
250 RETURN
400 LET k=x$: GO SUB 7300: REM
  move
410 IF nomat<>1 THEN GO SUB 7100:
  RETURN
420 IF relation=0 THEN LET z$="
  Which way to ?": GO SUB 7400: R
  ETURN
430 IF relation>2 THEN LET z$="
  I don't understand.": GO SUB 7400:
  RETURN
440 LET drn=(-1 AND relation=1)
  +(1 AND relation=2): LET from=fm
  at: IF y$<>"" THEN LET k=y$: G
  O SUB 7300: IF nomat<>1 THEN GO
  SUB 7100: RETURN: REM print ou
  t match error
450 LET to=o(fmat,4)+drn: IF to
  <1 OR to>8 THEN LET z$="There's
  no room to do that": GO SUB 7400:
  RETURN
452 FOR n=1 TO nosha: IF o(n,4)
  =to THEN LET z$="There's a"+FN
  a$(n)+" there already !": GO SUB
  7400: RETURN
455 IF o(n,5)=from THEN LET z$="
  What about the"+FN a$(n)+" on
  top !": GO SUB 7400: RETURN
460 NEXT n: LET wipe=1: LET pri
  ntx=from: GO SUB 8800: LET wipe=0:
  LET o(fmat,4)=to: LET o(fmat,5)=0:
  LET o(fmat,6)=0: GO SUB 8800:
  LET z$="O.k.": GO SUB 7400:
  RETURN
600 LET k=x$: GO SUB 7300: REM
  put
610 IF nomat<>1 THEN GO SUB 7100:
  RETURN
620 IF relation=0 THEN LET z$="
  What on ?": GO SUB 7400: RETURN
630 IF relation<>3 THEN LET z$=

```

```

="I don't understand.": GO SUB 7400:
  RETURN
640 LET from=fmat: LET k=y$: G
  O SUB 7300: IF nomat<>1 THEN GO
  SUB 7100: RETURN: REM print ou
  t match error
650 LET to=o(fmat,4): IF from=f
  mat THEN LET z$="You can't put
  a shape on itself !": GO SUB 7400:
  RETURN
652 IF o(fmat,3)>2 THEN LET z$="
  I won't balance !": GO SUB 7400:
  RETURN
655 FOR n=1 TO nosha: IF o(n,5)
  =from THEN LET z$="What about t
  he"+FN a$(n)+" on top !": GO SUB
  7400: RETURN
660 NEXT n: LET wipe=1: LET pri
  ntx=from: GO SUB 8800: LET wipe=0:
  LET o(fmat,4)=to: LET o(fmat,5)=0:
  LET o(fmat,6)=0: GO SUB 8800:
  LET z$="O.k.": GO SUB 7400:
  RETURN
800 LET k=x$: GO SUB 7300: REM
  where is
810 IF nomat<>1 THEN GO SUB 7100:
  RETURN
820 LET x=o(fmat,4): IF o(fmat,5)
  <>0 THEN LET z$="On top of th
  e"+FN a$(o(fmat,5)): GO SUB 7400:
  RETURN
830 LET w1=9: LET w2=9: LET n1=0:
  LET n2=0: LET under=0: REM fi
  nd nearest to left and right
835 FOR n=1 TO nosha
840 IF o(n,4)<x THEN IF x-o(n,4)
  <w1 THEN LET w1=x-o(n,4): LET

```



```

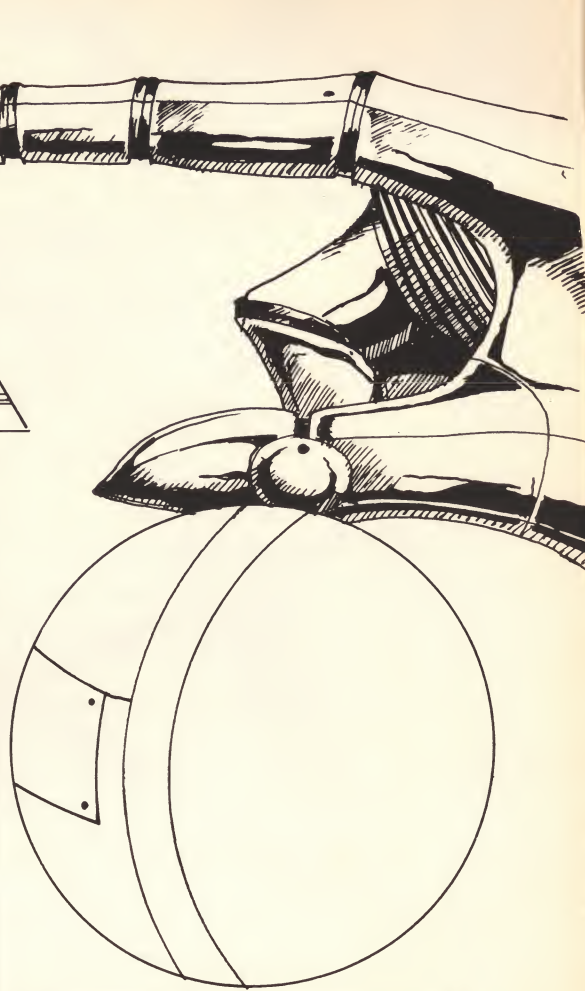
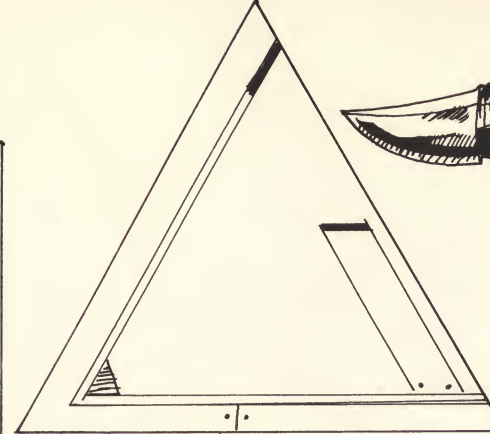
n1=n: REM nearest on left and r
  ight
860 IF o(n,5)=fmat THEN LET un
  der=n: REM underneath
865 NEXT n
870 IF n1 AND n2 THEN LET z$="
  It's between the"+FN a$(n1)+" an
  d the"+FN a$(n2)
880 IF n1 AND NOT n2 THEN LET
  z$="It's to the left of the"+FN
  a$(n1)
890 IF n2 AND NOT n1 THEN LET
  z$="It's to the right of the"+FN
  a$(n2)
900 IF under THEN LET z$=z$+"
  and" AND (n1 OR n2)+" under th
  e "+FN a$(under)
910 IF NOT (n1 OR n2 OR n2) THE
  N LET z$="It's the only shape"
920 GO SUB 7400: RETURN
1000 IF nosha=10 THEN LET z$="I
  can only make ten shapes": GO S
  UB 7400: RETURN: REM
  make
1010 LET space=1: LET n=1
1020 IF o(n,4)=space THEN LET s
  pace=space+1: LET n=1: GO TO 1020
1030 LET n=n+1: IF n<11 THEN GO
  TO 1020
1040 IF space=9 THEN LET z$="Th
  ere's no room to put in a new sha
  pe": GO SUB 7400: RETURN
1050 LET nosha=nosha+1: FOR n=1
  TO 6: LET o(nosha,n)=0: NEXT n:
  IF x$="" THEN GO TO 1100: REM n
  o parameters
1060 FOR n=1 TO LEN x$ STEP 2: L
  ET t=CODE x$(n)-3: LET nn=CODE x
  $(n+1): IF o(nosha,t)<>0 THEN L
  ET z$="That doesn't make sense":
  LET nosha=nosha-1: GO SUB 7400:
  RETURN: REM two same
1070 LET o(nosha,t)=nn: NEXT n
1100 IF o(nosha,1)=0 THEN LET z
  $="What size? Press 1 for small
  2 for large": GO SUB 7400: LET
  inmax=2: GO SUB 6900: LET o(nosh
  a,1)=inkey
1110 IF o(nosha,2)=0 THEN LET z
  $="What colour? Press 1 to 6 fo
  r blue to yellow": GO SUB 7400:
  LET inmax=6: GO SUB 6900: LET o(
  nosha,2)=inkey
1120 IF o(nosha,3)=0 THEN LET z
  $="What shape? Press 1 for a sq
  uare, 2 for an oblong, 3 for a
  triangle and 4 for a circle": G
  O SUB 7400: LET inmax=4: GO SUB
  6900: LET o(nosha,3)=inkey
1130 LET k$="": FOR n=1 TO 3: LE
  T k$=k$+CHR$(n+3)+CHR$(o(nosha,
  n)): NEXT n: GO SUB 7300: IF noma
  t>1 THEN LET z$="You'll confuse
  me if you have two shapes the s

```

```

ame": LET nosha=nosha-1: GO SUB
  7400: RETURN
1140 LET o(nosha,4)=space: LET w
  ipe=0: LET printx=nosha: GO SUB
  8800: LET z$="O.k.": GO SUB 7400
  : RETURN
1200 LET k=x$: GO SUB 7300: REM
  destroy
1210 IF nomat<>1 THEN GO SUB 7100:
  RETURN: REM not unique
1220 FOR n=1 TO nosha: IF o(n,5)
  =fmat THEN LET z$="What about t
  he shape on top ?": GO SUB 7400:
  RETURN
1230 NEXT n
1240 LET printx=fmat: LET wipe=1
  : GO SUB 8800: REM print it out
1249 REM now move the others dow
  n to fill gap (if is one)
1250 IF fmat<>nosha THEN FOR n=
  1 TO 6: LET o(fmat,n)=o(nosha,n)
  : NEXT n
1260 FOR n=1 TO nosha: IF o(n,5)
  =nosha THEN LET o(n,5)=fmat
1270 NEXT n
1280 LET nosha=nosha-1: LET z$="
  O.k.": GO SUB 7400: RETURN
1400 LET z$="Are you sure you wa
  nt to quit? (y/n)": GO SUB 7400
  0: REM
  quit
1410 LET i$=INKEY$: IF i$<>"y" A
  ND i$<>"n" THEN GO TO 1410
1420 IF i$="n" THEN RETURN
1430 CLS: PRINT "Type 'RUN' t
  hen Enter to restart": STOP
1600 REM
  help
1610 LET z$="In this program you
  can instruct the computer to ma
  nipulate the objects above on th
  e screen. The computer will try
  to give intelligent answers. For
  example you could type in 'move
  the red circle right of the gre
  en triangle'. For a list of the
  words understood type commands
  ": GO SUB 7400: RETURN
1800 REM
  commands
1810 LET z$="I understand the wo
  rds list, move, put, make, where
  is, destroy, left, right, on an
  d the names and colours of the s
  hapes."
1820 LET z$=z$+" Some examples a
  re... 'move the green triangle
  left' 'where is the small circ
  le' 'list the red shapes' 'pu
  t the circle on the square'.
  "+CHR$(17)+CHR$(6)+"Press any
  key for more.": GO SUB 7400
1830 IF INKEY$="" THEN GO TO 1830
1840 LET z$="You can also abbrev
  iate all of the commands and wor
  ds down to their first three let
  ters. For example 'mov gre tri l
  ef of red tri' would be interpre
  ted as 'move green triangle left
  of red triangle'. To finish ty
  pe in quit.": GO SUB 7400: RETU
  R N
1999 REM
  ** the input and analyse **
2000 LET x$="": LET y$="": LET a
  ction=0: LET rel=0: LET end=0: G
  O SUB 7900: REM get input

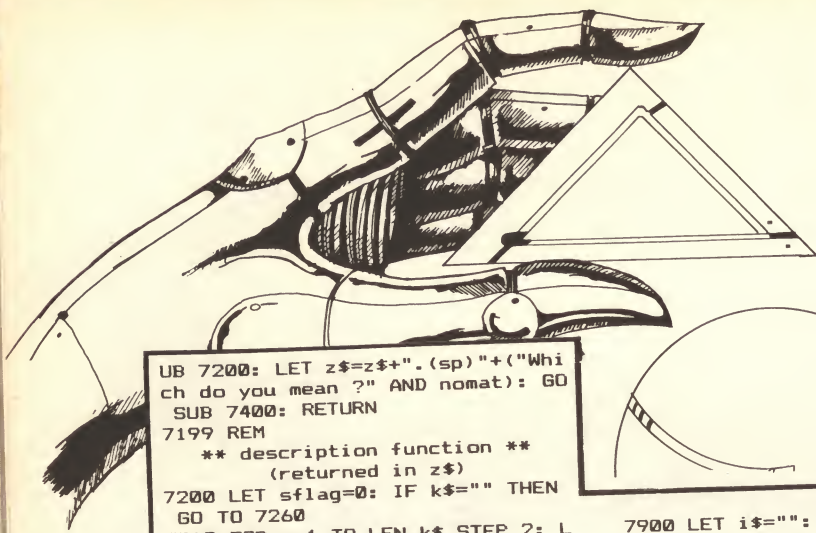
```



```

2010 GO SUB 7800: REM first word
  in w$, i$ TO
2020 IF end THEN GO TO 2000: RE
  M no action
2030 GO SUB 7700: IF end THEN G
  O SUB 7600: GO TO 2000
2040 LET action=number: IF actio
  n=7 THEN RETURN: REM all done
2045 REM now check for ignores
2050 GO SUB 7800: IF end THEN R
  ETURN
2060 GO SUB 7700: IF end THEN G
  O SUB 7600: GO TO 2000
2070 IF type=3 THEN GO TO 2050:
  REM skip ignores
2080 IF type<4 THEN GO SUB 7600
  : GO TO 2000: REM expect a decri
  ptor
2090 LET x$=x$+CHR$(type)+CHR$(n
  umber)
2100 GO SUB 7800: IF end THEN R
  ETURN
2110 GO SUB 7700: IF end THEN G
  O SUB 7600: GO TO 2000
2120 IF type=4 THEN GO TO 2090
  : REM next descriptor
2130 IF type=3 THEN GO TO 2100
2140 IF type=1 THEN GO SUB 7600
  : GO TO 2000:
2150 IF type=2 THEN LET relatio
  n=number
2160 GO SUB 7800: IF end THEN R
  ETURN
2170 GO SUB 7700: IF end THEN G
  O SUB 7600: GO TO 2000
2180 IF type=3 THEN GO TO 2160
2190 IF type<4 THEN GO SUB 7600
  : GO TO 2000: REM 2 relatives
2200 LET y$=y$+CHR$(type)+CHR$(n
  umber): GO TO 2160
6899 REM inkey$ (1 to inmax)
  returned in inkey.....
6900 LET g$=INKEY$: IF g$<"1" OR
  g$>STR$(inmax) THEN GO TO 6900
6910 LET inkey=VAL g$: RETURN
6999 STOP
7100 LET z$="I can't" AND no
  mat=0)+" see"+FN q$(nomat): GO S

```

```

UB 7200: LET z$=z$+" (sp)" + ("Whi
ch do you mean?" AND nomat): GO
SUB 7400: RETURN
7199 REM

```

```

** description function **
(returned in z$)

```

```

7200 LET sflag=0: IF k$="" THEN
GO TO 7260

```

```

7210 FOR n=1 TO LEN k$ STEP 2: L
ET t=CODE k$(n)-3: LET nn=CODE k
$(n+1)

```

```

7220 IF t=1 THEN LET z$=z$+FN s
$(nn)

```

```

7230 IF t=2 THEN LET z$=z$+FN c
$(nn)

```

```

7240 IF t=3 THEN LET z$=z$+FN n
$(nn): LET sflag=1

```

```

7250 NEXT n
7260 IF sflag=0 THEN LET z$=z$+
" shape"

```

```

7270 IF nomat<>1 THEN LET z$=z$
+"s"

```

```

7280 RETURN
7299 REM ** match function **

```

```

7300 REM f(10) all set by this
1=match 0=no-match

```

```

noofmatches & firstmatch
also set

```

```

7302 REM hold 0 if free -1 if c
onflicting conditions else cond

```

```

7303 REM uses k$ to pass params
(x$/y$)

```

```

7310 FOR n=1 TO nosha: LET f(n)=
1: NEXT n: LET nomat=nosha: LET
fmat=1

```

```

7320 IF k$="" THEN RETURN
7330 FOR n=1 TO LEN k$ STEP 2: L
ET t=CODE k$(n)-3: LET nn=CODE k
$(n+1): FOR m=1 TO nosha: IF f(m
)=1 THEN IF o(m,t)<nn THEN LE
T f(m)=0: LET nomat=nomat-1

```

```

7340 NEXT m: NEXT n
7350 IF f(fmat)=1 THEN RETURN

```

```

7360 LET fmat=fmat+1: IF fmat<1
1 THEN GO TO 7350

```

```

7370 LET fmat=0: RETURN: REM no
ne

```

```

7399 REM ** print z$ **
7400 PRINT #1;AT 0,0;

```

```

7410 IF LEN z$<33 THEN PRINT 1
;AT 0,0;z$: LET v=USR USR "o": R
ETURN

```

```

7420 FOR l=32 TO 10 STEP -1: IF
z$(l)<>"(sp)" THEN NEXT l: LET
l=32

```

```

7430 PRINT #1;AT 0,0;z$( TO l):
LET v=USR USR "o": LET z$=z$(l+1
TO ): GO TO 7400

```

```

7599 REM ** error routine **
7600 LET v=USR USR "o": LET z$="
I don't understand ": IF type=0
THEN LET z$=z$+w$

```

```

7610 GO SUB 7400: RETURN
7699 REM ** find word in dict**

```

```

7700 RESTORE 9000+CODE w$(1)-COD
E ("a"): LET w$=w$+"(3*sp)"

```

```

7710 READ z$,type,number: IF z$=
"(3*sp)" THEN LET end=1: RETURN

```

```

7720 IF w$( TO 3)<>z$ THEN GO T
O 7710

```

```

7730 RETURN
7800 LET w$="": IF i$="" THEN L
ET end=1: RETURN

```

```

7810 IF i$(1)="(3*sp)" THEN LET i$=i
$(2 TO ): GO TO 7800

```

```

7820 LET n=1
7830 IF i$(n)<>"(sp)" AND n<LEN
i$ THEN LET n=n+1: GO TO 7830

```

```

7835 IF n=LEN i$ THEN LET w$=i$
: LET i$="": RETURN

```

```

7840 LET w$=i$( TO n-1): LET i$=
i$(n TO ): RETURN

```

```

7900 LET i$="": PRINT #1;AT 0,0;
">";

```

```

7910 IF INKEY$<>"(sp)" THEN GO TO 7
910

```

```

7920 LET z$=INKEY$: IF z$="" THE
N GO TO 7920

```

```

7930 IF z$="(sp)" AND z$="(3*sp)" AN
D LEN (i$)<59 THEN LET i$=i$+z$
: PRINT #1;AT 0,1;i$;"(g4,sp)";:
GO TO 7910

```

```

7940 IF z$=CHR$ 13 THEN PRINT #
1;AT 0,1;i$;"(sp)";: LET v=USR U
SR "o": RETURN

```

```

7950 IF z$=CHR$ 12 AND LEN (i$)>
0 THEN LET i$=i$( TO LEN i$-1):
PRINT #1;AT 0,1;i$;"(g4,sp)";:
GO TO 7910

```

```

7960 BEEP .1,12: GO TO 7910
7999 REM set up variables

```

```

8000 DIM o(20,6): DIM f(10): LET
nosha=5: REM objects + matches

```

```

8001 REM 6th= height
8010 REM
1. size 2.colour

```

```

3. name 4.x pos
5. status (0=ground)
6. height above gnd

```

```

8020 REM size small or large
colour red/blue/green
/yellow (1-4)

```

```

8030 REM name square/oblong/
triangle/circle

```

```

8040 REM status 0=on ground
x=on object no x

```

```

7900 LET i$="": PRINT #1;AT 0,0;
">";

```

```

7910 IF INKEY$<>"(sp)" THEN GO TO 7
910

```

```

7920 LET z$=INKEY$: IF z$="" THE
N GO TO 7920

```

```

7930 IF z$="(sp)" AND z$="(3*sp)" AN
D LEN (i$)<59 THEN LET i$=i$+z$
: PRINT #1;AT 0,1;i$;"(g4,sp)";:
GO TO 7910

```

```

7940 IF z$=CHR$ 13 THEN PRINT #
1;AT 0,1;i$;"(sp)";: LET v=USR U
SR "o": RETURN

```

```

7950 IF z$=CHR$ 12 AND LEN (i$)>
0 THEN LET i$=i$( TO LEN i$-1):
PRINT #1;AT 0,1;i$;"(g4,sp)";:
GO TO 7910

```

```

7960 BEEP .1,12: GO TO 7910
7999 REM set up variables

```

```

8000 DIM o(20,6): DIM f(10): LET
nosha=5: REM objects + matches

```

```

8001 REM 6th= height
8010 REM
1. size 2.colour

```

```

3. name 4.x pos
5. status (0=ground)
6. height above gnd

```

```

8020 REM size small or large
colour red/blue/green
/yellow (1-4)

```

```

8030 REM name square/oblong/
triangle/circle

```

```

8040 REM status 0=on ground
x=on object no x

```

```

8050 RESTORE 8060: FOR n=1 TO 5:
FOR m=1 TO 6: READ o(n,m): NEXT
m: NEXT n

```

```

8060 DATA 1,1,1,1,0,0: REM small
blue square

```

```

8070 DATA 2,2,3,3,0,0: REM large
red triangle

```

```

8080 DATA 1,2,4,4,0,0: REM small
red circle

```

```

8090 DATA 2,3,2,7,0,0: REM large
green oblong

```

```

8100 DATA 1,4,3,7,4,4: REM small
yellow triangle on the oblong

```

```

8200 LET wipe=0: FOR m=1 TO 5: L
ET printx=m: GO SUB 8800: NEXT m

```

```

8500 RETURN
8800 REM print in an object (x)

```

```

8810 LET px=o(printx,4)*4-4: LET
py=o(printx,6): REM: IF o(prin
tx,4)<>0 THEN LET

```

```

py=1+o(o(printx,4),3)
8820 LET pc=o(printx,2)

```

```

8830 LET psize=o(printx,1): REM
size

```

```

8840 RESTORE 8870+2*o(printx,3)+
o(printx,1)

```

```

8850 FOR n=py TO py+psize+o(prin
tx,3)=2: READ p$: IF wipe=1 TH
EN LET p$="(6*sp)" ( TO LEN p$)

```

```

8860 PRINT AT 9-n,px; INK pc;p$:
NEXT n: RETURN

```

```

8873 DATA "(2*ig8)", "(2*ig8)"
8874 DATA "(3*ig8)", "(3*ig8)", "(
3*ig8)"

```

```

8875 DATA "(2*ig8)", "(2*ig8)", "(
2*ig8)"

```

```

8876 DATA "(3*ig8)", "(3*ig8)", "(
3*ig8)", "(3*ig8)"

```

```

8877 DATA "LM", "JK"
8878 DATA "L(ig8)", "J(ig8)", "
L(ig8)", "J(ig8)"

```

```

8879 DATA "L(ig8)", "J(ig8)", "
L(ig8)", "J(ig8)"

```

```

8880 DATA "L(ig8)", "J(ig8)", "
L(ig8)", "J(ig8)"

```

```

8881 DATA "L(ig8)", "J(ig8)", "
L(ig8)", "J(ig8)"

```

```

8882 DATA "L(ig8)", "J(ig8)", "
L(ig8)", "J(ig8)"

```

```

8883 DATA "L(ig8)", "J(ig8)", "
L(ig8)", "J(ig8)"

```

```

8884 DATA "L(ig8)", "J(ig8)", "
L(ig8)", "J(ig8)"

```

```

8885 DATA "L(ig8)", "J(ig8)", "
L(ig8)", "J(ig8)"

```

```

8886 DATA "L(ig8)", "J(ig8)", "
L(ig8)", "J(ig8)"

```

```

8887 DATA "L(ig8)", "J(ig8)", "
L(ig8)", "J(ig8)"

```

```

8888 DATA "L(ig8)", "J(ig8)", "
L(ig8)", "J(ig8)"

```

```

8889 DATA "L(ig8)", "J(ig8)", "
L(ig8)", "J(ig8)"

```

```

8890 DATA "L(ig8)", "J(ig8)", "
L(ig8)", "J(ig8)"

```

```

8891 DATA "L(ig8)", "J(ig8)", "
L(ig8)", "J(ig8)"

```

```

8879 DATA "CD", "BB"
8880 DATA "E(ig8)", "(3*ig8)", "(
ig8)", "(3*ig8)"
8999 RESTORE 9000: FOR n=1 TO 10
0: READ a$,b,c: PRINT a$,b,c: NE
XT n
9000 DATA "a(2*sp)", "3,2", "(3*0)",
0,0
9001 DATA "bal", "6,4", "blu", "5,1", "b
ig", "4,2", "(3*0)", "0,0: REM b
9002 DATA "cir", "6,4", "cya", "5,5", "c
om", "1,9", "(3*0)", "0,0: REM c
9003 DATA "des", "1,6", "3*", "0,0: R
EM d
9004 REM e
9005 REM f
9006 DATA "gre", "5,4", "(3*0)", "0,0:
REM g
9007 DATA "hel", "1,8", "(3*0)", "0,0:
REM h
9008 DATA "is", "3,5", "(3*0)", "0,0:
REM i
9009 REM j
9010 REM k
9011 DATA "lis", "1,1", "lef", "2,1", "l
ar", "4,2", "(3*0)", "0,0: REM l
9012 DATA "mov", "1,2", "mak", "1,5", "m
ag", "5,5", "0,0: REM m
9013 REM n
9014 DATA "obl", "6,2", "off", "2,4", "o
n(sp)", "2,3", "ont", "2,3", "of(sp)", "3,
4", "(3*0)", "0,0: REM o
9015 DATA "put", "1,3", "(3*0)", "0,0:
REM p
9016 DATA "qui", "1,7", "(3*0)", "0,0:
REM q
9017 DATA "rec", "6,2", "red", "5,2", "r
ig", "2,2", "(3*0)", "0,0: REM r
9018 DATA "squ", "6,1", "sma", "4,1", "s
ha", "3,5", "(3*0)", "0,0: REM s
9019 DATA "tri", "6,3", "to(sp)", "3,1
", "the", "3,3", "(3*0)", "0,0: REM t
9020 REM u
9021 REM v
9022 DATA "whe", "1,4", "wha", "1,1", "(
3*0)", "0,0: REM w
9023 REM x
9024 DATA "yel", "5,6", "(3*0)", "0,0:
REM y
9025 DATA "(3*0)", "0,0: REM z
9100 REM 1 list 2 move 3 put
4 where5 make 6 destroy
7 quit 8 help 9
9200 DATA 15,63,127,127,255,255,
255,255: REM A
9210 DATA 240,252,254,254,255,25
5,255,255: REM B
9220 DATA 255,255,255,255,127,12
7,63,15: REM C
9230 DATA 255,255,255,255,254,25
4,252,240: REM D
9240 DATA 0,7,31,63,63,127,127,1
27: REM E
9250 DATA 0,224,248,252,252,254,
254,254: REM F
9260 DATA 127,127,127,63,63,31,7
,0: REM G
9270 DATA 254,254,254,252,252,24
8,224,0: REM H
9280 DATA 24,24,60,60,126,126,25
5,255: REM I
9290 DATA 1,1,3,3,7,7,15,15: REM
J
9300 DATA 128,128,192,192,224,22
4,240,240: REM K
9310 DATA 31,31,63,63,127,127,25
5,255: REM L
9320 DATA 248,248,252,252,254,25
4,255,255: REM M
9330 DATA 255,129,66,60,60,66,12
9,255: REM N
9340 DATA 6,12,205,0,14,201,0,0:
REM O (carefull this is a m/c
routine
9380 DATA 6,12,205,0,14,201,0,0:
REM u (carefull this is a m/c
routine
9400 CLS: PRINT AT 3,10;"A.I. P
rogram.": AT 5,2;"Poking in data.
Please wait.": RESTORE 9200: FO
R n=0 TO 127: READ a: POKE USR "
a"+n,a: NEXT n: RETURN
9997 STOP
9998 GO SUB 9400: RUN
9999 SAVE "a.i." LINE 9998

```

QUADRUN

AVOID the objects and the computer generated tracks in **Quadrun** by Tat Tang of Aylesbury in Buckinghamshire. The game, which runs on the 16K Spectrum, uses a very pretty effect to make the screen more attractive, and IN instructions for reading the keyboard instead of the more conventional INPUT. Use keys Q and Z to move up and down, G and H for left and right.

Quadrun uses our special abbreviations for graphics characters, so please read the instructions on the first page of Program Printout before typing in the program.

```

5 LET HI=0: GO TO 1000
10 INK 1: PLOT 255-X,175-Y
20 PLOT 255-X,Y
30 PLOT X,175-Y
35 IF RND>.9 THEN GO SUB 200
40 IF POINT (X,Y)=1 THEN GO T
O 300
45 PLOT INK 2;X,Y
50 LET A=-(IN 65278<>255 OR IN
32766<>255)+(IN 64510<>255 OR I
N 57342<>255)
55 LET S=S+1
60 LET B=(49150<>255)-(65022<>
255)
63 IF IN 65022<>255 THEN LET
B=-1
65 IF IN 49150<>255 THEN LET
B=1
70 IF A=0 AND B=0 THEN GO TO
90
80 LET G=A: LET H=B
90 LET X=X+H: LET Y=Y+G
95 IF H=0 THEN LET H=(RND>.5)
-(RND<.5)
97 IF G=0 THEN LET G=(RND>.5)
-(RND<.5)
100 IF X=-1 THEN LET X=255
110 IF X=256 THEN LET X=0
120 IF Y=-1 THEN LET Y=175
130 IF Y=176 THEN LET Y=0
135 IF S>HI THEN LET HI=S
140 PRINT AT 22,0;"SCORE=";S;AT

```

```

22,16;"HI=";HI: GO TO 10
200 LET C=RND*11: LET D=RND*16
210 PRINT AT C,D; FLASH 1; INK
RND*4;"(ig6)";AT 21-C,31-D;"(ig6
)"; INVERSE 1;AT 21-C,D;"(ig6)";
AT C,31-D;"(ig6)": RETURN
300 PRINT AT 11,14; FLASH 1;"DE
AD": FOR N=30 TO -36 STEP -3: BE
EP .03,N: NEXT N
310 IF INKEY$<>"(sp)" THEN GO TO 3
10
320 IF INKEY$="" THEN GO TO 32
0
1000 POKE 23659,2: POKE 23693,57
: CLS
1010 PRINT "Briefing:"""The Yea
r: 2001""The Place: The Matrix"
""The Game: Quadrun""The Prize:
Survival"
1020 PRINT ""1)Cover as much as
possible"
1030 PRINT "2)Avoid Computer Gen
erated Trax"
1050 PRINT ""Press any key to pl
ay The Game"
1060 PAUSE 0
2000 LET X=INT (RND*127): LET Y=
INT (RND*78)
2010 LET G=1: LET H=0: LET A=0:
LET B=0
2020 LET S=0: CLS: POKE 23659,1
: GO TO 10

```


FIGHT YOUR WAY through hordes of monsters and collect all the crystals in a superb graphics adventure by Jacek Przychodzen of South Australia. **Crystal Quest** for the 48K Spectrum is in the tradition of adventures such as **Halls of the Things**, with a randomly generated dungeon for you to explore. Collect the crystals, and use the objects you find to aid you.

The program uses our special abbreviations for graphics characters so please read the instructions on the first page of Program Printout.

```
1 BEEP 1,50: CLEAR: DIM s(16)
: LET pt=0: LET pit=450: LET d$="THE CHEST CONTAINS ": GO SUB 5000: BORDER 0: INK 7: PAPER 0: CLS: GO SUB 7000: GO SUB 650: PRINT #0: INK 5: BRIGHT 1: "Do you want to load your (8*sp)previous character? (Y/N)"
```

```
2 IF INKEY$="Y" THEN GO TO 8500
```

```
3 IF INKEY$=CHR$(110) THEN GO TO 10
```

```
4 BEEP .005,0: GO TO 2
```

```
10 LET knc=0: LET ke1=0: LET ke=0: LET cro1=3: LET cro2=5: LET cro3=8: LET cro4=40: LET cg=0: LET ms=0: LET diam=0: LET st=0: LET co=0: LET gol=0: LET en=10: LET ro=1: LET sc=0: LET m=0
```

```
11 LET wb=0: LET m$="": LET ps=215: LET k6=5: LET k5=177: LET k4=70: LET k3=6: LET k2=69: LET k1=68: LET x=15: LET y=12
```

```
20 LET cc=0: LET mh=0: LET mo=0: LET mv=0: GO SUB 8050: LET sc=sc+(INT(st/5)): FOR l=0 TO 31: PRINT AT 1,1: INK 4: PAPER 6: "a"NEXT 1
```

```
21 LET st=st+50: IF st<99 THEN LET st=99
```

```
22 FOR l=2 TO 20: PRINT AT 1,0: INK 4: PAPER 6: "a": AT 1,31: "a"NEXT 1
```

```
23 PRINT AT 1,16: BRIGHT 1: INK 7: "L": AT 21,16: "L": AT 11,0: "L": AT 11,31: "L"
```

```
24 GO SUB 9999: IF ro=cro3 THEN GO TO 39
```

```
25 LET rn=INT(RND*10)+1
26 FOR l=1 TO rn: RESTORE 38: FOR b=1 TO INT(RND*10)+1: READ a: NEXT b: GO SUB a: NEXT 1: GO TO 39
```

```
27 FOR n=1 TO 10: PRINT AT 12,n: INK 4: PAPER 6: "a": NEXT n: RETURN
```

```
28 FOR n=4 TO 15: PRINT AT 16,n: INK 4: PAPER 6: "a": NEXT n: RETURN
```

```
29 FOR n=9 TO 19: PRINT AT 8,n: INK 4: PAPER 6: "a": NEXT n: RETURN
```

```
30 FOR n=6 TO 25: PRINT AT 4,n: INK 4: PAPER 6: "a": NEXT n: RETURN
```

```
31 FOR n=20 TO 30: PRINT AT 10,n: INK 4: PAPER 6: "a": NEXT n: RETURN
```

```
32 FOR n=6 TO 17: PRINT AT n,7: INK 4: PAPER 6: "a": NEXT n: RETURN
```

```
33 FOR n=6 TO 18: PRINT AT n,26: INK 4: PAPER 6: "a": NEXT n: RETURN
```

```
34 FOR n=2 TO 6: PRINT AT n,15: INK 4: PAPER 6: "a": NEXT n: RETURN
```

```
35 FOR n=10 TO 20: PRINT AT n,17: INK 4: PAPER 6: "a": NEXT n: RETURN
```

```
36 FOR n=6 TO 17: PRINT AT n,22: INK 4: PAPER 6: "a": NEXT n: RETURN
```

CRYSTAL QUEST

RETURN

```
38 DATA 27,28,29,30,31,32,33,34,35,36
```

```
39 GO SUB 260: PRINT AT y,x: FLASH 1: INK 5: "u": GO SUB 600
```

```
40 IF INKEY$<">" THEN LET st=st-1: GO SUB 9999: GO SUB 500
```

```
41 IF mo=0 THEN LET mo=1: GO SUB 9990: GO SUB 9991: GO SUB 1000: LET my=yy: LET mx=xx: GO SUB 641: PRINT AT my,mx: INK 2: BRIGHT 1: m$
```

```
42 IF mo=1 THEN LET mc=2: LET mb=1: GO SUB 1010
```

```
45 IF ATTR(mcy,mcx)=52 THEN LET sc=sc-1: GO SUB 640
```

```
52 IF x=0 OR x=31 OR y=1 OR y=21 THEN BEEP .2,2: BEEP .2,2: BEEP .2,2: BEEP .3,10: BEEP .2,5: GO SUB 250: LET sc=sc+5: LET ro=ro+1: CLS: GO TO 20
```

```
199 GO TO 40
```

```
220 REM *** firing ***
```

```
221 LET xc=x+1: FOR l=xc TO xc+10: BEEP .01,-10: IF SCREEN$(y,1)="(sp)" AND l<31 THEN PRINT AT y,1: INK 7: "K": GO SUB 2000: NEXT l
```

```
223 FOR q=xc TO l-1: PRINT AT y,q: "K": NEXT q: GO SUB 3010: RETURN
```

```
230 LET xc=x-1: FOR l=xc TO xc-
```

```
10 STEP -1: BEEP .01,-10: IF SCREEN$(y,1)="(sp)" AND l>0 THEN PRINT AT y,1: INK 7: BRIGHT 1: "K"
```

```
"": GO SUB 2005: NEXT 1
```

```
231 FOR q=xc TO l+1 STEP -1: PRINT AT y,q: "K": NEXT q: GO SUB 3010: RETURN
```

```
235 LET yc=y+1: FOR l=yc TO yc+10: BEEP .01,-10: IF SCREEN$(1,x)="(sp)" AND l<21 THEN PRINT AT 1,x: INK 7: BRIGHT 1: "K": GO SUB 2008: NEXT 1
```

```
236 FOR q=yc TO l-1: PRINT AT q,x: "K": NEXT q: GO SUB 3010: RETURN
```

```
240 LET yc=y-1: FOR l=yc TO yc-10 STEP -1: BEEP .01,-10: IF SCREEN$(1,x)="(sp)" AND l>1 THEN PRINT AT 1,x: INK 7: BRIGHT 1: "K": GO SUB 2010: NEXT 1
```

```
241 FOR q=yc TO l+1 STEP -1: PRINT AT q,x: "K": NEXT q: GO SUB 3010: RETURN
```

```
250 IF x=0 THEN LET wb=3: LET x=30
```

```
251 IF x=31 THEN LET wb=1: LET x=1
```

```
252 IF y=21 THEN LET wb=2: LET y=2
```

```
253 IF y=1 THEN LET wb=4: LET y=20
```

```
254 IF y=2 AND pt=1 THEN LET w
```

```
b=4: LET y=20
255 RETURN
```

```
260 IF wb=3 THEN PRINT AT 11,31: INK 4: PAPER 6: "a"
```

```
261 IF wb=1 THEN PRINT AT 11,0: INK 4: PAPER 6: "a"
```

```
262 IF wb=2 THEN PRINT AT 1,16: INK 4: PAPER 6: "a"
```

```
263 IF wb=4 THEN PRINT AT 21,16: INK 4: PAPER 6: "a"
```

```
265 RETURN
```

```
300 IF en=0 THEN PRINT AT 1,0: PAPER 7: INK 2: FLASH 1: "OUT OF AMMO": FOR f=1 TO 4: BEEP .1,40: NEXT f: PRINT AT 1,0: INK 4: PAPER 6: "(13*a)": RETURN
```

```
301 LET en=en-1
```

```
302 IF m=2 THEN GO SUB 235
```

```
303 IF m=4 THEN GO SUB 240
```

```
304 IF m=1 THEN GO SUB 220
```

```
305 IF m=3 THEN GO SUB 230
```

```
306 RETURN
```

```
400 IF RND>.9 THEN LET ke=ke+INT(RND*3)+1: LET n$="WHITE KEY": GO SUB 9996: PRINT AT cy,cx: INK 7: BRIGHT 1: "a": GO TO 540
```

```
405 IF RND>.7 THEN LET n$="COIN": GO SUB 9996: PRINT AT cy,cx: "a": GO TO 540
```

```
410 IF RND>.5 THEN LET n$="AMMO": LET en=en+INT(RND*3)+1: GO SUB 9996: PRINT AT cy,cx: INK
```

```
5: BRIGHT 1: "N": GO TO 540
```

```
415 IF RND>.2 THEN LET n$="A COIN": GO SUB 9996: PRINT AT cy,cx: INK 6: "a": GO TO 540
```

```
417 IF RND>.1 THEN LET n$="ELIXIR": GO SUB 9996: PRINT AT cy,cx: INK 4: BRIGHT 1: "a": GO TO 540
```

```
420 LET n$="A BLUE KEY": GO SUB 9996: PRINT AT cy,cx: INK 1: BRIGHT 1: "a": GO TO 540
```

```
425 PRINT AT 15,6: PAPER 7: INK 1: "Another go? (Y/N)"
```

```
426 IF INKEY$="Y" THEN BEEP .5,30: RUN
```

```
427 IF INKEY$="N" THEN BEEP .5,0: STOP
```

```
428 GO TO 426
```

```
450 LET px=0: LET py=0: LET mc=4: LET mb=0: LET m$="a": LET pt=1: CLS: GO SUB 490: PRINT #0: FLASH 1: INK 7: PAPER 2: "SNAKE PIT"
```

```
451 FOR l=0 TO 31: PRINT AT 2,1: INK 4: PAPER 6: "a": AT 21,1: "a": NEXT 1: FOR l=3 TO 20: PRINT AT 1,0: INK 4: PAPER 6: "a": AT 1,31: "a": IF l<13 THEN PRINT AT 1,2: INK 4: PAPER 6: "a"
```

```
452 NEXT 1: PRINT INK 3: BRIGHT 1: AT 6,1: "L": AT 2,1: "L": AT 12,1: "L": AT 4,1: INK 7: "L": AT 9,1: "L": AT 4,29: INK 4: BRIGHT 1: "a": AT 19,29: "a"
```

```
453 PRINT AT y,x: INK 5: "u": FOR l=1 TO 2: GO SUB 9990: PRINT AT yy,xx: INK 7: BRIGHT 1: "a": NEXT 1
```

```
454 FOR l=1 TO 3: GO SUB 9990: PRINT AT yy,xx: INK 1: BRIGHT 1: "a": NEXT 1
```

```
455 IF INKEY$<">" THEN GO SUB 490: LET st=st-1: GO SUB 9999: GO SUB 500
```

```
456 IF y=2 THEN LET x=15: FOR l=7 TO 0 STEP -1: BEEP .02,1: PRINT AT my,mx: BRIGHT 1: INK 1: m$: NEXT 1: BEEP .2,2: BEEP .2,2: BEEP .2,2: BEEP .3,10: BEEP .2,5: GO SUB 250: LET sc=sc+20: LET ro=ro+1: CLS: LET pt=0: GO TO 20
```

```
458 IF mo=0 THEN LET mo=1: GO SUB 9990: GO SUB 1000: LET m$="a": LET my=yy: LET mx=xx: PRINT AT my,mx: INK mc: BRIGHT mb: m$
```

```
460 GO SUB 1010
```

```
461 IF mx=3 THEN IF my<14 THEN GO SUB 640
```

```
480 GO TO 455
```

```
490 LET h$="(sp)": IF ke1>1 THEN LET h$="S"
```

```
495 LET g$="(sp)": IF ke>1 THEN LET g$="S"
```

```
496 PRINT AT 1,0: "KEY"+g$: "a" (sp): "ke": "KEY"+h$: INK 1: BRIGHT 1: "a" (sp): INK 7: BRIGHT 0: ke1
```

```
499 RETURN
```

```
500 IF INKEY$="q" THEN LET m=4: LET cy=y-1: LET cx=x: GO SUB 510: IF SCREEN$(cy,cx)="(sp)" THEN PRINT AT y,x: "a": LET y=y-1: PRINT AT y,x: INK 5: "u"
```

```
501 IF INKEY$="a" THEN LET m=2: LET cy=y+1: LET cx=x: GO SUB 510: IF SCREEN$(cy,cx)="(sp)" THEN PRINT AT y,x: "a": LET y=y+1: PRINT AT y,x: INK 5: "u"
```

```
502 IF INKEY$="p" THEN LET m=1: LET cy=y: LET cx=x+1: GO SUB 510: IF SCREEN$(cy,cx)="(sp)" THEN PRINT AT y,x: "a": LET x=x+1: PRINT AT y,x: INK 5: "u"
```

```
503 IF INKEY$="o" THEN LET m=3: LET cy=y: LET cx=x-1: GO SUB 510: IF SCREEN$(cy,cx)="(sp)" THEN PRINT AT y,x: "a": LET x=x-1: PRINT AT y,x: INK 5: "u"
```

```
505 IF INKEY$="(sp)" THEN IF c g=1 THEN GO SUB 300
```

```
506 IF st<=0 THEN PRINT AT 8,9: BRIGHT 1: FLASH 1: INK 6: PAPER 2: "ADVENTURE LOST": GO TO 425
```

```
507 IF px=x THEN IF py=y AND ro<40 THEN FOR l=1 TO 21: BEEP .005,1*2: LET f=USR 3280: NEXT 1: GO TO pit
```

```
508 RETURN
```

```
510 IF ATTR(cy,cx)>7 THEN GO SUB 515
```

```
511 RETURN
```

```
515 IF ATTR(cy,cx)=k1 THEN LET T sc=sc+5: LET st=st+100: GO SUB 9998: LET n$="ELIXIR": GO SUB 9995: GO SUB 9999: RETURN
```

```
516 IF ATTR(cy,cx)=k2 THEN LET T n$="AMMUNITION": LET sc=sc+5: LET en=en+4: GO SUB 9998: GO SUB 9995: GO SUB 9999: RETURN
```

```
517 IF ATTR(cy,cx)=k4 THEN LET T sc=sc+50: LET gol=gol+1: GO SUB 9998: LET n$="GOLD BARS": GO SUB 9995: GO SUB 9999: RETURN
```

```
518 IF ATTR(cy,cx)=k3 THEN LET T sc=sc+2: LET co=co+1: GO SUB 9998: LET n$="COIN": GO SUB 9995: GO SUB 9999: RETURN
```

```
519 IF ATTR(cy,cx)=k5 THEN LET T sc=sc+500: LET n$="CRYSTAL": LET diam=diam+1: GO SUB 9998: GO SUB 9995: GO SUB 9999: RETURN
```

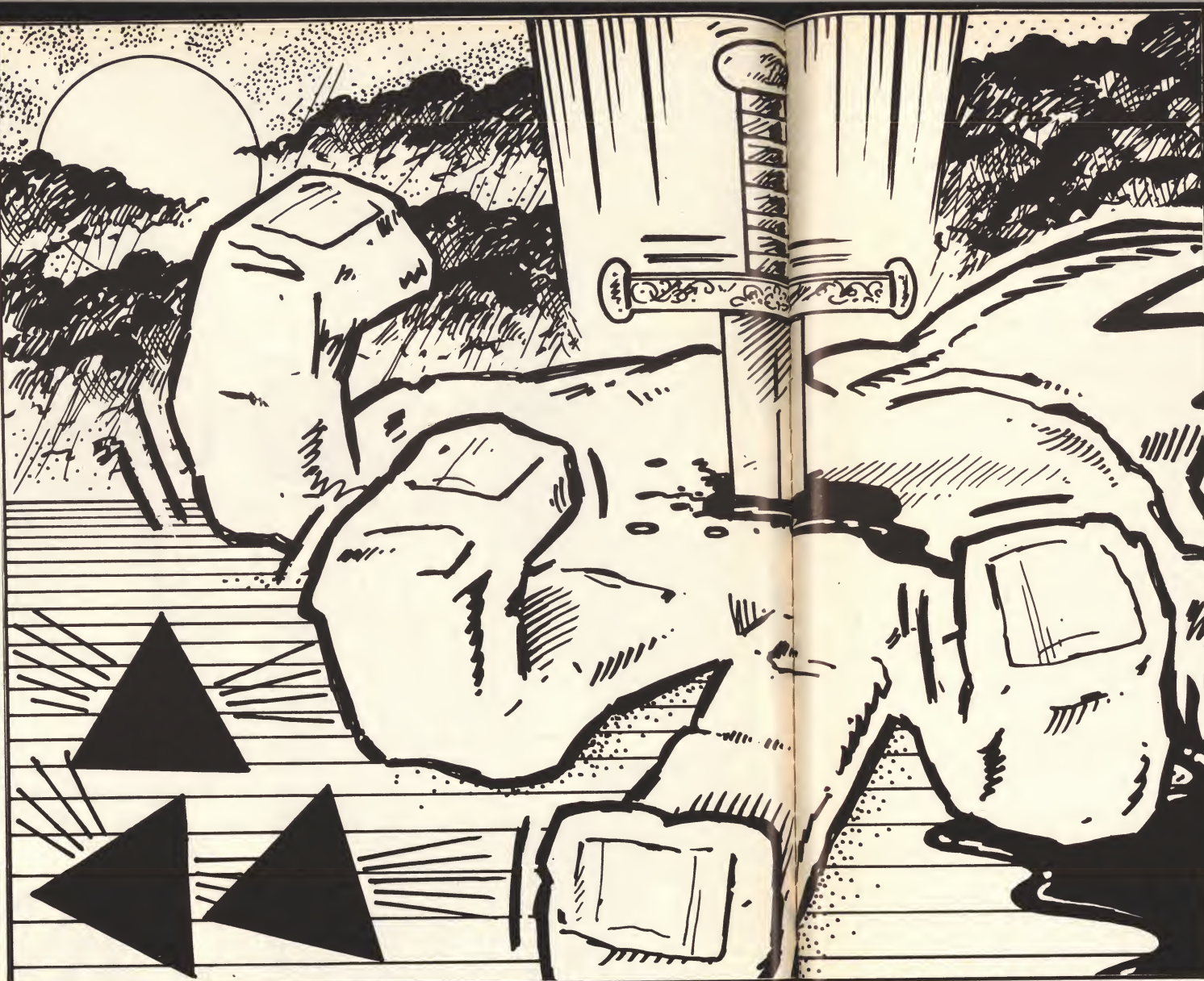
```
520 IF ATTR(cy,cx)=k6 THEN LET
```



```

T n$="(3*sp,iG,iU,iN,sp)": LET c
g=1: LET sc=sc+6: LET en=en+1: G
O SUB 9998: GO SUB 9995: BEEP .1
,5: BEEP .1,7: BEEP .1,5: BEEP .
2,0: GO SUB 9999: RETURN
521 IF ATTR (cy,cx)=71 AND pt=0
THEN IF cx>0 AND cx<31 AND cy>
1 AND cy<21 THEN LET sc=sc+2: L
ET ke=ke+1: GO SUB 9998: LET n$=
" WHITE KEY ": GO SUB 9995: GO S
UB 9999: RETURN
522 IF ATTR (cy,cx)=71 AND ke>0
THEN IF cx<2 OR cx=31 OR cy=21
OR cy=1 THEN GO SUB 640: LET s
c=sc+5: LET ke=ke-1: FOR f=-50 T
O -30: BEEP .05,f: BEEP .04,f-5:
NEXT f: PRINT AT cy,cx: PAPER 0
;"(sp)": GO SUB 9999: RETURN
523 IF ATTR (cy,cx)=67 AND ke1>
0 THEN GO SUB 640: LET sc=sc+4:
LET ke1=ke1-1: GO SUB 9999: FOR
f=-50 TO -30: BEEP .1,f: NEXT f
: PRINT AT cy,cx;"(sp)": RETURN
524 IF ATTR (cy,cx)=65 THEN PR
INT AT cy,cx;"(sp)": LET ke1=ke1
+1: LET sc=sc+8: LET n$=" BLUE K
EY ": GO SUB 9995: BEEP .1,10: B
EEP .1,6: GO SUB 9999: RETURN
525 IF ATTR (cy,cx)=2 THEN LET
sc=sc+2: GO SUB 9999: BEEP .01,
-10: GO TO 400
527 IF ATTR (cy,cx)=psi THEN F
OR l=1 TO 30: PRINT AT y,x: INK
INT (RND*6)+1: BRIGHT RND;"_": B
EEP .001,RND*50: NEXT l: GO TO 4
25
528 IF ATTR (cy,cx)=71 AND pt=1
THEN IF cx>0 AND cx<31 AND cy>
1 AND cy<21 THEN LET sc=sc+2: L
ET ke=ke+1: GO SUB 9998: LET n$=
" WHITE KEY ": GO SUB 9995: GO S
UB 9999: RETURN
540 RETURN
600 LET b$=" HALLWAY ROOM ": IF
ro=cro1 THEN LET cro1=ro+3: GO
SUB 700: GO SUB 705: LET b$=" S
UPPLY ROOM "
601 IF ro=cro2 THEN LET cro2=r
o+3: GO SUB 710: GO SUB 9990: PR
INT AT yy,xx: INK 1; BRIGHT 1;"_
": LET b$=" VAULT ROOM "
602 IF ro=cro3 THEN LET cro3=r
o+8: GO SUB 720: LET b$=" CRYSTA
L ROOM "
603 IF diam>5 AND ro>40 THEN
GO TO 8000
604 IF diam<5 AND ro>140 THEN
PRINT AT 10,6: INK 6; PAPER 2;
BRIGHT 1;"ADVENTURE LOST"; FLASH
0; INK 7; PAPER 0; AT 12,1;"YOU
DIDN'T GET ALL CRYSTALS": FOR l=
1 TO 5: FOR s=1 TO 20: BEEP .01,
s: NEXT s: NEXT l: GO SUB 8050:
STOP
605 IF RND>.7 THEN GO SUB 9990
: PRINT AT yy,xx: INK 2;"_":
606 IF RND>.9 THEN GO SUB 9990
: PRINT AT yy,xx: INK 1; BRIGHT
1;"_":
618 IF cg=0 THEN GO SUB 9990:
GO SUB 9991: PRINT AT yy,xx: INK
5; PAPER 0; BRIGHT 0;"M":
619 GO SUB 9990: GO SUB 9991: P
RINT AT yy,xx: INK 6; PAPER 0;"_
":
620 PRINT#0: INK 5;" ROOM No. "
;ro;"(sp)": PAPER 7; INK 1;b$:
621 GO SUB 9990: PRINT AT yy,xx
: INK 7; BRIGHT 1;"_":
624 GO SUB 9990: LET py=yy: LET
px=xx
625 FOR l=1 TO 30: IF SCREEN$ (
py,l)<>"(sp)" THEN NEXT l
626 PRINT AT py,l: INK 4;"_": B
EEP .001,50: PRINT AT py,l: PAPE
R 0;"(sp)": NEXT l
629 DEF FN s()=INT (10-(mv*10))
*10
630 RETURN
640 LET mo=0: FOR l=7 TO 0 STEP
-1: PRINT AT my,mx: BRIGHT 1; I

```



```

NK 1;m$: BEEP .02,1: NEXT l: PRI
NT AT my,mx: PAPER 0;"(sp)": RET
URN
641 FOR l=0 TO 7: BEEP .02,1: P
RINT AT my,mx: BRIGHT 1; INK 1;m
$: NEXT l: RETURN
650 PRINT#0:"Instructions ?(Y/
N)"
651 IF INKEY$=CHR$ (110) THEN
BEEP .5,10: RETURN
652 IF INKEY$="y" THEN BEEP .5
,20: GO TO 655
653 GO TO 651
655 CLS : PRINT AT 1,9; PAPER 7
: INK 2; BRIGHT 1;"INSTRUCTIONS"
656 PRINT : INK 5; BRIGHT 1;" T
he object of the game is to(3*sp
)complete successfully a dungeon
generated randomly by the(7*sp)
computer."
657 PRINT INK 5; BRIGHT 1;" On
the way you must collect(4*sp)"
: PAPER 7; INK 2; FLASH 1;"5"; F
LASH 0; INK 5; PAPER 0;" crystal
s.To pick up additional points c
ollect other objects"
658 PRINT BRIGHT 1;" To pick o
bject move your man to one of th
e objects sides and(3*sp)press d
irection key."
659 PRINT BRIGHT 1;" You can o
nly shoot when you(4*sp)carry a
gun "; INK 5;"(M)" INK 7;"and a
mmunition "; INK 4;"(N)."
660 PRINT BRIGHT 1; INK 2;" Yo
u will lose game only when" INK
6;" -1 You step into"(4*sp)ps
i-field "; FLASH 1; INK 7; BRIGH
T 1; PAPER 2;"RRRRRR"
661 PRINT BRIGHT 1;" -2 You'll
run out of strength" INK 2;" W
hen monster hits you it will(2*s
p)disarm you and will take away(

```

```

3*sp)some of your strength."#0;
INK 7;" Press any key"
662 PAUSE 0: BEEP .2,0: CLS
663 PRINT BRIGHT 1;"You can d
eactivate psi-field by shooting
at the generators from the dista
nce of exactly one" "character"
664 PRINT TAB (3); INK 4; PAPER
2;"R"; PAPER 0; INK 5;"-ACTIVE
GENERATOR"
665 PRINT TAB (3); INK 5; PAPE
R 1; FLASH 1;"R"; PAPER 0; FLASH
0; INK 5;"-INACTIVE GENERATOR"
666 PRINT BRIGHT 1;"Blue key";
INK 1;"(Q)"; INK 7;"opens magen
ta"; INK 3;"(L)"; INK 7;"door"
667 PRINT BRIGHT 1;"White key(
Q)opens white (L)door"
668 PRINT BRIGHT 1;"You can't
reuse any of the keys"
669 PRINT INK 4;"Upon entering
a room snake will run across in
dicating trapdoor(2*sp)to the sn
ake pit."
670 PRINT INK 1; PAPER 7;" CHR
$ Name(7*sp)score"
671 PRINT BRIGHT 1; INK 2;"(2
*sp)I"; INK 6;" Mutant E.T.30pts
"
672 PRINT BRIGHT 1; INK 2;"(2
*sp)H"; INK 6;" Goggles.....20pts
"
673 PRINT BRIGHT 1; INK 2;"(2
*sp)G"; INK 6;" Mlot.....10pts
"
674 PRINT INK 4;"(2*sp)Q"; IN
K 3;" Snake.....N/A "; INK 7;"c
an't kill it"
675 PRINT#0: INK 7;"Press any
key": PAUSE 0: BEEP .2,0: CLS
679 PRINT PAPER 7; INK 1;"In y
our quest you'll find;"
680 PRINT INK 4;"E"; INK 6;" -

```

```

Elixir.It will increase your";T
AB (11);"strength and improve";T
AB (11);"your final score"
681 PRINT INK 2;"_"; INK 6;" -
Chest.It can contain";TAB (11);
"many useful";TAB (11);"objects"
682 PRINT INK 6; BRIGHT 1;"_";
INK 6;" - Coin.Found in every r
oom.";TAB (11);"It will increase
your";TAB (11);"final score."
683 PRINT INK 6; BRIGHT 1;"_";
INK 6;" - Gold.Found in vault r
oom."
684 PRINT INK 1; PAPER 6; FLAS
H 1; BRIGHT 1;"F"; PAPER 0; FLAS
H 0; INK 6;" - Crystal.Found in
Treasure";TAB (11);"room.To get
it you'll";TAB (11);"have to kno
ck out";TAB (11);"psi-field"
685 PRINT INK 5; BRIGHT 1;"M";
INK 6;" - Gun.(2*sp)Found in ro
om next to";TAB (11);"the one wh
ere monster";TAB (11);"attacked
you last."
686 PRINT INK 4; BRIGHT 1;"N";
INK 6;" - Ammo. Found in supply
room."
687 PRINT INK 7; BRIGHT 1;"_";
INK 6;" - W.Key. Found in every
room."
688 PRINT INK 1; BRIGHT 1;"_";
INK 6;" - B.Key. Found in some
rooms.";TAB (11);"Or in boxes."
689 PRINT#0: INK 7;"Press any
key": PAUSE 0: BEEP .2,0: CLS
690 PRINT AT 1,1; INK 1; PAPER
7;" MOVEMENT CONTROLS "
691 PRINT " "; INK 6; BRIGHT 1;T
AB (10);"Q"; INK 5;" - UP" TAB
(5); INK 6;"O"; INK 5;" - LEFT "
; INK 6;"P"; INK 5;" - RIGHT" TAB
AB (10); INK 6;"A"; INK 5;" - DO
WN"

```

```

692 PRINT INK 4;" Press "; IN
K 6;"SPACE"; INK 4;" to fire in
the(5*sp)direction of the last m
ove."
693 PRINT " Once you'll leave ro
om and your strength will be low
er than 100 it will be automatic
ly raised(3*sp)back to 100."
694 PRINT FLASH 1; PAPER 7; IN
K 2; BRIGHT 1;"IMPORTANT"; FLASH
0; PAPER 0; INK 5;" - To leave
room you(3*sp)";TAB (12);"must c
ollect white ";TAB (12);"key fir
st!"
695 PRINT INK 7; BRIGHT 1;"- I
f you wish to save your(8*sp)pro
gress on tape press S in(5*sp)st
atus mode."
699 PRINT#0: INK 7;"Press any
key to start.": PAUSE 0: BEEP .2
,10: CLS : RETURN
700 GO SUB 9990
702 PRINT AT yy,xx: INK 4; PAPE
R 0; BRIGHT 1;"E"
703 RETURN
705 GO SUB 9990: GO SUB 9991: P
RINT AT yy,xx: INK 5; PAPER 0; B
RIGHT 1;"N": RETURN
710 PRINT AT 10,12; INK 4; PAPE
R 6;"(6*sp)";AT 14,12;"(6*sp)";AT
12,12; INK 3; BRIGHT 1;"_";AT 12
,18;"_":
711 FOR l=11 TO 13: PRINT AT 1,
12; INK 4; PAPER 6;"R"; PAPER 0;
"(5*sp)"; PAPER 6;"R": NEXT l
712 PRINT AT 11,15; INK 6; BRIG
HT 1;"E"
713 PRINT AT 12,12; INK 3; BRIG
HT 1;"_";AT 12,18;"_":
719 RETURN
720 FOR l=8 TO 23: PRINT AT 6,1
; INK 4; PAPER 6;"R";AT 16,1;"R"
; NEXT l
721 FOR l=10 TO 21: PRINT AT 8,
1; INK 4; PAPER 6;"R";AT 14,1;"R
": NEXT l
723 FOR l=7 TO 15: PRINT AT 1,8
; INK 4; PAPER 6;"R";AT 1,23;"R"
: NEXT l
724 FOR l=9 TO 13: PRINT AT 1,1
0; INK 4; PAPER 6;"R";AT 1,21;"R
": NEXT l
725 PRINT AT 6,15; INK 3; BRIGH
T 1;"_";AT 14,15;"_";AT 10,11; I
NK 4; PAPER 2;"R";AT 10,20;"R";A
T 10,12; FLASH 1; INK 7; BRIGHT
1; PAPER 2;"RRRRRRRR"
726 LET gh=0
727 PRINT AT 9,INT (RND*10)+11;
INK 1; PAPER 6; FLASH 1;"F"
728 PRINT AT 3,29; INK 1; BRIGH
T 1;"_";AT 19,2;"_":
730 RETURN
1000 IF RND<.4 THEN LET m$="G":
LET mv=.8: RETURN
1001 IF RND<.8 THEN LET m$="H":
LET mv=.6: RETURN
1002 LET m$="I": LET mv=.4: RETU
RN
1010 IF my>y THEN LET mcx=my-1:
LET mcx=mx: IF SCREEN$ (mcx,mcx
)="(sp)" THEN PRINT AT my,mx;"(
sp)": LET my=my-1: PRINT AT my,m
x; INK mc; BRIGHT mb;m$: RETURN
1011 IF my<y THEN LET mcx=my+1:
LET mcx=mx: IF SCREEN$ (mcx,mcx
)="(sp)" THEN PRINT AT my,mx;"(
sp)": LET my=my+1: PRINT AT my,m
x; INK mc; BRIGHT mb;m$: RETURN
1012 IF mx>x THEN LET mcx=mx: L
ET mcx=mx-1: IF SCREEN$ (mcx,mcx
)="(sp)" THEN PRINT AT my,mx;"(
sp)": LET mx=mx-1: PRINT AT my,m
x; INK mc; BRIGHT mb;m$: RETURN
1013 IF mx<x THEN LET mcx=mx: L
ET mcx=mx+1: IF SCREEN$ (mcx,mcx
)="(sp)" THEN PRINT AT my,mx;"(
sp)": LET mx=mx+1: PRINT AT my,m
x; INK mc; BRIGHT mb;m$: RETURN
1014 IF mcx=y AND mcx=x THEN GO
SUB 640: PRINT AT my,mx: PAPER

```

```

0;"(sp)": LET cg=0: LET mo=0: LE
T sc=INT (sc-FN s()): GO SUB 999
9: FOR l=-10 TO 50 STEP 3: PRINT
AT y,x: INK 7; BRIGHT 1;"_": BE
EP .01,1+5: PRINT AT y,x: INK 2;
BRIGHT 1;"_": BEEP .01,1: NEXT
l: PRINT AT y,x: PAPER 0; INK 5;
"_": IF st>10 THEN LET st=INT (
st-FN s()): GO SUB 9999
1015 IF ATTR (mcx,mcx)=215 THEN
GO SUB 640
1020 RETURN
1030 LET de=.001: PRINT AT 1-1,x
; INK 5; PAPER 1; FLASH 1;"R";AT
1,x; FLASH 0; INK 7; PAPER 0;"(
sp)": LET sc=sc+5: FOR l=30 TO 5
0: BEEP .001,1: NEXT l: LET gh=
gh+1
1031 IF gh=2 THEN FOR n=1 TO 4:
FOR f=7 TO 0 STEP -1: PRINT AT
10,12; INK f; BRIGHT 1;"(8*sp)";
BEEP de,f*6: NEXT f: NEXT n: PRI
NT AT 10,12; PAPER 0;"(8*sp)"
1032 RETURN
1040 LET de=.001: PRINT AT 1-1,x
; INK 5; PAPER 1; FLASH 1;"R";AT
1,x; FLASH 0; INK 7; PAPER 0;"(
sp)": LET sc=sc+10: FOR l=30 TO 5
0: BEEP .001,1: NEXT l: LET gh=
gh+1
1041 IF gh=2 THEN LET gh=0: FOR
n=1 TO 4: FOR f=7 TO 0 STEP -1:
PRINT AT y-2,2; INK f; BRIGHT 1
;"(28*sp)"; BEEP de,f*8: NEXT f:
NEXT n: PRINT AT y-2,2; PAPER 0;
"(28*sp)": FOR f=7 TO 0 STEP -1:
PRINT AT y-2,1; INK f; PAPER 0;
"R";AT y-2,30;"R": BEEP .001,f*8
: NEXT f: PRINT AT y-2,1; INK 7;
"(sp)";AT y-2,30;"(sp)"
1045 RETURN
1060 REM *** end of prog. ***
1061 PRINT AT 11,2; PAPER 7; INK
1;" THIS GAME IS NOW COMPLETED
"
1062 PRINT AT 13,2; INK 6;"Your
total score is.."; INK 5;sc
1063 PRINT AT 14,2; INK 2;"Coins
collected.....";co
1064 PRINT AT 15,2; INK 4;"Coin
bonus.....";cp*5
1065 PRINT AT 17,21; INK 3;" ____
";AT 18,10; INK 5;"TOTAL SCO
RE >";
1066 PRINT AT 16,2; INK 5;"Stren
ght.....";st: LET cob=
sc: FOR l=0 TO st*2 STEP 9: PRIN
T AT 18,23;cob+l: BEEP .01,30: N
EXT l: LET sc=sc+1
1067 LET cob=sc: FOR l=0 TO co*5
STEP 9: BEEP .01,55: PRINT AT 1
8,23;cob+l: NEXT l: LET sc=sc+1
1068 PRINT AT 19,1; INK 7; BRIGH
T 1;"Press any key to see status
";AT 20,1;" and save results."
1070 IF INKEY$="" THEN GO TO 10
70
1071 BEEP .2,0: LET knc=1: GO TO
8050
2000 IF ATTR (y,l+1)=66 THEN GO
SUB 3015
2001 RETURN
2005 IF ATTR (y,l-1)=66 THEN GO
SUB 3015
2006 RETURN
2008 IF ATTR (l+1,x)=66 THEN GO
SUB 3015
2009 RETURN
2010 IF ATTR (l-1,x)=66 THEN GO
SUB 3015
2012 IF ATTR (l-1,x)=84 THEN IF
ro+20=cro3 THEN GO SUB 1030
2013 IF ATTR (l-1,x)=84 THEN IF
ro<cro3 AND ATTR (y-2,x)=84 TH
EN GO SUB 1040
2015 RETURN
3000 LET ms=ms+1: LET sc=sc+(FN
s()/2): GO SUB 9999: LET mo=0: R
ETURN
3010 IF mh=1 THEN PRINT AT my,m
x; PAPER 0;"(sp)": LET mh=0: GO
SUB 3000: RETURN

```



```

3011 RETURN
3015 LET mh=1: BEEP .05,50: FOR
f=7 TO 0 STEP -1: PRINT AT my,mx
: INK f: BRIGHT 1:"P": BEEP .01,
(f*10)/2: NEXT f: RETURN
4001 PRINT s$: LET s1=VAL s$(LEN
s$-2 TO LEN s$): PRINT s1*100
4050 STOP
5000 RESTORE 5001: FOR 1=USR "a"
TO USR "u"+7: READ a: POKE 1,a:
NEXT 1
5001 DATA 0,247,247,247,0,127,12
7,127
5002 DATA 0,8,8,8,BIN 11100,BIN
11110,BIN 11100,0
5003 DATA 0,BIN 11111,66,BIN 11
11101,BIN 11001101,BIN 10110101
,BIN 11001110,BIN 11111100
5004 DATA 0,0,0,0,BIN 11100,BIN
11110,BIN 11100,0
5005 DATA 0,0,28,28,0,BIN 111011
1,BIN 111011,0
5006 DATA 0,8,BIN 11100,BIN 1010
10,BIN 1001001,BIN 101010,BIN 11
100,8
5007 DATA 8,BIN 10100,BIN 111110
,BIN 101101,BIN 1001001,BIN 111
00,BIN 10100,BIN 10100
5008 DATA 0,BIN 10100,BIN 101010
,BIN 11100,BIN 111110,BIN 101110
1,8,BIN 11100
5009 DATA BIN 111100,BIN 11000,B
IN 11000,BIN 111110,BIN 1011010
,BIN 11000,BIN 100100,BIN 110011
0
5010 DATA BIN 111000,40,BIN 1111
100,BIN 1011010,BIN 111000,68,4
0,BIN 1101100
5011 DATA 0,0,0,24,24,0,0,0
5012 DATA 60,66,BIN 10011001,BIN
10100101,BIN 10100111,BIN 10100
101,BIN 10100101,255
5013 DATA 0,0,BIN 11111,BIN 1110
100,BIN 11100100,BIN 11000100,0,
0
5014 DATA 0,8,28,28,20,8,28,0
5015 DATA 0,0,0,BIN 1100001,BIN
10010101,BIN 10011111,BIN 110000
0,0
5016 DATA 0,8,34,0,BIN 1001001,0
,34,8
5017 DATA 8,BIN 11100,32,16,8,BI
N 100100,BIN 1000100,BIN 111000
5018 DATA 102,153,102,153,102,15
3,102,153
5019 DATA BIN 11000,16,BIN 11100
0,BIN 110110,BIN 110000,8,16,BIN
110000
5020 DATA BIN 11000,8,BIN 11100,
BIN 1101100,BIN 1100,8,16,BIN 11
000
5021 DATA BIN 111000,BIN 111010,
BIN 1111010,BIN 10111100,BIN 101
11000,68,40,BIN 1101100
5022 RETURN
7000 LET c$="(32*sp)": PRINT AT
0,0: INK 2:"(sp,30*ig3)"
7001 PRINT AT 21,0: INK 2:"(sp,3
0*ig3)"
7002 FOR 1=1 TO 20: PRINT AT 1,0
: INK 2:"(g5)":AT 1,31:"(ig5)":
NEXT 1
7003 PRINT AT 19,13: INK 7: BRIG
HT 1:"1984": PRINT #0:"(9*sp)Pre
ss any key"
7005 FOR 1=1 TO 7
7006 PRINT AT 4,3: INK 1:"(ig6,2
*ig3,sp,ig4,g3,g6,sp,ig5,sp,g5,sp
,ig4,2*ig3,sp,g3,g7,g3,sp,ig6,g3,
g6,sp,ig5)"
7007 PRINT AT 5,3: INK 1:"(ig5,3*
sp,ig1,ig3,ig6,sp,g6,ig3,ig6,sp,
2*ig3,g7,2*sp,g5,2*sp,ig1,ig3,ig2
,sp,ig5)"
7008 PRINT AT 6,3: INK 1:"(g6,ig
3,sp,ig5,sp,g6,2*sp,g5,2*sp,2*ig
3,ig2,2*sp,g5,2*sp,ig5,sp,g5,sp,
ig1,ig3,ig7)"
7009 PRINT AT 9,7: INK 1:"(ig6,g
3,g6,sp,ig5,sp,g5,sp,ig6,2*ig3,sp
,ig4,2*ig3,sp,g3,g7,g3)"
7010 PRINT AT 10,7: INK 1:"(ig5,
ig7,g5,sp,ig5,sp,g5,sp,ig4,g3,2*
sp,2*ig3,g7,2*sp,g5)"

```

```

7011 PRINT AT 11,7: INK 1:"(g6,i
g2,ig6,sp,g6,ig3,ig6,sp,g6,2*ig3
,sp,2*ig3,ig2,2*sp,g5)"
7012 PRINT AT 13,6: INK 8-1:"DES
IGNED AND WRITTEN"
7013 PRINT AT 17,7: INK 8-1:"JAC
EK PRZYCHODZEN"
7014 PRINT AT 15,14: INK 8-1:"BY
7088 IF INKEY$<>" THEN GO TO 7
091
7089 NEXT 1
7090 IF INKEY$="" THEN GO TO 70
05
7091 BEEP .5,0: CLS : RETURN
8000 PRINT #0:"ROOM No.":ro: INK
2: FLASH 1: BRIGHT 1: PAPER 6:"
EXIT ROOM": LET gh=0: LET x=15:
LET y=20: PRINT AT 11,0: INK 4:
PAPER 6:"E":AT 21,16:"E":AT 11,3
1:"E": FOR 1=10 TO 20: PRINT AT
1,1:"(30*sp)": NEXT 1: PRINT AT
19,INT (RND*27)+2: INK 5:"M"
8001 FOR 1=1 TO 8: PRINT AT 1,0:
PAPER 1:"(32*sp)": NEXT 1: PRIN
T AT 9,0: INK 4: PAPER 6:"(32*E)
":AT 9,15: INK 3: BRIGHT 1: PAPE
R 0:"L": INK 6: FOR 1=-1 TO -3 S
TEP -1: PLOT 232+1,135+1: DRAW -
15,15,-PI: NEXT 1: INK 7
8002 FOR 1=10 TO 110 STEP 22: IN
K 4: PLOT 1,104: DRAW 0,5: DRAW
-7,0: DRAW 10,25: DRAW 10,-25: D
RAW -7,0: DRAW 0,-5
8003 LET kk=-8: LET ff=10: FOR f
=111 TO 133 STEP 2: PLOT 1+3,f:
DRAW ff,0: LET ff=ff-kk: NEXT f:
LET ff=10: NEXT 1: INK 7: FOR 1
=1 TO 8: PRINT AT INT (RND*4)+1,
INT (RND*25): INK 7: PAPER 1:"."
: NEXT 1
8004 FOR 1=11 TO 19 STEP 3: PRIN
T AT 1,1: INK 4: PAPER 2: BRIGHT
1:"R": INK 7: FLASH 1:"(28*R)":
FLASH 0: INK 4:"R": NEXT 1
8005 IF INKEY$<>" THEN LET st=
st-1: GO SUB 9999: GO SUB 500
8006 IF y=9 THEN PRINT AT y,x:"
(sp)": LET y=y-1: FOR 1=14 TO 30
: PRINT AT 8,1: PAPER 1:"(sp)J":
BEEP .001,1*2: NEXT 1: PRINT AT
8,31: PAPER 1:"(sp)": GO TO 106
0
8007 IF mo=0 THEN LET cc=0: LET
mo=1: GO SUB 1000: LET my=20: L
ET mx=INT (RND*28)+1: GO SUB 641
: PRINT AT my,mx: INK 2: BRIGHT
1:m$
8008 LET cc=cc+1: IF cc=2 THEN
LET cc=0: LET mc=2: LET mb=1: GO
SUB 1010
8009 IF cg=0 THEN PRINT AT 19,1
5: INK 5:CHR$ (156)
8010 GO TO 8005
8050 CLS
8051 DRAW 255,0: DRAW 0,175: DRA
W -255,0: DRAW 0,-175
8052 PLOT 3,3: DRAW 249,0: DRAW
0,170: DRAW -249,0: DRAW 0,-170
8055 PRINT AT 1,6: INK 4: BRIGHT
.1:"* * * STATUS * * *"
8058 PRINT AT 4,2: INK 6:"EXPERI
ENCE..":sc:AT 6,2: INK 4: BRIGHT
1:"ROOM.....":ro:AT 8,2: INK
3:"STRENGTH....":st
8059 PRINT AT 10,2: INK 5:"SHOTS
LEFT ": FOR f=12 TO INT 11+en:
IF f<31 THEN PRINT AT 10,f: INK
2: BRIGHT 1:"N"
8060 BEEP .001,40: NEXT f: PRINT
AT 11,1: INK 2: BRIGHT 1:"-----
"
8061 PRINT AT 12,4: PAPER 1: INK
6:"CRYSTALS ":diam:AT 14,4:"GOL
D(5*sp)":gol:AT 16,4:"COIN(5*sp)
":co
8062 IF cg=1 THEN PRINT AT 2,1:
INK 2: PAPER 7:"You are armed"
8063 IF cg=0 THEN PRINT AT 2,1:
INK 2: PAPER 7: FLASH 1:"You ar
e unarmed"
8064 PRINT AT 13,18: INK 7:"KEY(
s)": INK 1: BRIGHT 1:"(sp)": I
NK 7:ke1:AT 15,18:"KEY(s)": INK
7:"(sp)":ke

```

```

8065 PRINT AT 17,1: INK 6:"-----
":AT 18,2: INK 2:"M
ONSTERS SLAIN ":ms
8100 PRINT #0: INK 7: BRIGHT 1:"
(5*sp)Press (S) to save game"
8200 PRINT AT 20,1: INK 2: FLASH
1: PAPER 6:" PRESS ANY KEY TO C
ONTINUE ": IF knc=1 THEN PRINT
AT 20,1: PAPER 7: INK 1:"(7*sp)G
AME FINISHED(7*sp)"
8201 IF INKEY$="s" THEN GO TO 8
300
8202 IF INKEY$="" THEN GO TO 82
01
8203 IF knc=0 THEN : CLS : BEEP
.2,10: RETURN
8204 STOP
8300 REM ***** SAVE game *****
8301 CLS : PRINT AT 10,6: FLASH
1: INK 1: PAPER 5:" INSERT DATA
CASSETTE ": LET s(1)=st: LET s(2)
=en: LET s(3)=diam: LET s(4)=gol
: LET s(5)=cg
8305 LET s(6)=ro
8306 LET s(7)=sc
8307 LET s(8)=cro1
8308 LET s(9)=cro2
8309 LET s(10)=cro3
8310 LET s(11)=cro4
8311 LET s(12)=co
8312 LET s(13)=ms
8313 LET s(14)=ke: LET s(15)=ke1
: LET s(16)=knc
8325 INPUT " Name of the charact
er ?(8*sp) ( max 10 letters ) ":x
$
8330 SAVE x$ DATA s()
8335 GO TO 8050
8500 REM ***** READ game *****
8501 CLS : INK 6: INPUT "Charact
er's name ?":x$: IF LEN x$>10 TH
EN BEEP 1,55: GO TO 8501
8502 PRINT AT 20,3: BRIGHT 1: FL
ASH 1: INK 7: PAPER 1:" PRESS AN
Y KEY TO LOAD ":AT 18,1:"INSERT
DATA-CASSETTE AND"
8503 PAUSE 0: BEEP .5,0: CLS : P
RINT AT 21,0: FLASH 1: BRIGHT 1:
" LOADING ": IF x$="" THEN PRI
NT "something":
8504 PRINT FLASH 0:"(sp)":x$: L
OAD x$ DATA s()
8505 LET ro=s(6)
8506 LET sc=s(7)
8507 LET cro1=s(8)
8508 LET cro2=s(9)
8509 LET cro3=s(10)
8510 LET cro4=s(11)
8511 LET co=s(12)
8512 LET ms=s(13)
8513 LET knc=s(16): LET ke=s(14)
: LET ke1=s(15): LET st=s(1): LE
T en=s(2): LET diam=s(3): LET go
l=s(4): LET cg=s(5)
8515 CLS : PRINT FLASH 1:AT 10,
5:"STOP THE TAPE":AT 12,10:"GAME
LOADED": FOR 1=0 TO 60: BEEP .0
1,10: NEXT 1: CLS
8520 INK 7: BEEP .3,10: BEEP .2,
0: GO TO 11
9990 LET yy=INT (RND*17)+3: LET
xx=INT (RND*27)+3
9991 IF ATTR (yy,xx)<>7 THEN GO
TO 9990
9992 RETURN
9993 REM UDG's
A=A B=B C=C D=D E=E F=F
G=G H=H I=I J=J K=K L=L
M=M N=N O=O P=P Q=Q R=R
S=S T=T U=U
9995 PRINT AT cy,cx: PAPER 0:"(s
p)":AT 0,0: FLASH 1: INK 6: PAPE
R 2:"(3*sp)COLLECTED(3*sp)":n$:
FOR 1=0 TO 40: BEEP .01,1: NEXT
1: PRINT AT 0,0:c$: RETURN
9996 PRINT AT 0,0:c$: PRINT AT c
y,cx: PAPER 0:"(sp)":AT 0,0: INK
5:d$+n$: PAUSE 20: FOR 1=45 TO
-10 STEP -1: BEEP .01,1: NEXT 1
9998 PRINT AT 0,0:c$: RETURN
9999 PRINT AT 0,0:"EXPERIENCE ":
sc:" STRENGTH ":st:"(sp)": RETUR
N

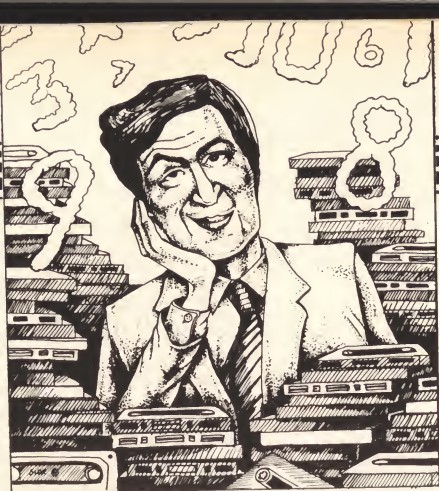
```

Chapter 12

Reference



What Gilbert Factor was The Turk awarded?
 Who publishes it?
 Where can I find it?
 We furnish the answers in our Software Directory.
 In addition we tell you where to go
 when your Spectrum blows up
 and list the growing number
 of Sinclair-related clubs
 at home and abroad.



If we have reviewed a program we have given it a rating known as the Gilbert Factor, named after our Software editor, John Gilbert. That factor includes clarity of instructions, speed of operation, ease of use, originality, lasting appeal, use made of graphics and sound, and success in accomplishing stated aims.

ZX-81

Adventure

Adventure	Bug-Byte	6
Adventure 1	Abersoft	6
Black Crystal	Mastervision	7
Dungeons of Doom	Woosoft	*
Espionage Island	Artic	*
Greedy Gulch	Phipps	7
Inca Curse	Sinclair	*
Knight's Quest	Phipps	*
Lost Island	JRS	*
Magic Mountain	Phipps	7
Merchant of Venus	Crystal	*
Pharaoh's Tomb	Phipps	*
Pimania	Automata	6
Secret Valley	Newsoft	*
Serpents Tomb	Vortex	*
Ship of Doom	Artic	7
The Great Western	New Soft	*
Time Bandits	New Soft	*
Tomb of Dracula	Felix	*
Trader Trilogy	Quicksilva	6
Volcanic Dungeon	Mastervision	5
World of Illusions	Contrast	4

Arcade

Alien Dropout	Silversoft	*
Asteroids	Silversoft	8
Asteroids	Software Farm	6
Astral Convoy	Vortex	*
Bank Robber	Romik	6
Bears in the Wood	Unicorn	6
Bubble Bugs	Romik	6
Byter	Protek	*
Cassette 1	Orwin	7
Cassette 2-5	Orwin	7
City Patrol	Sinclair	7
Damper	Quicksilva	7
Defenda	Quicksilva	8
Door Slammer	Cathedral	7
Forty-Niner	Software Farm	9
Froggy	DJL	7
Full-screen Breakout	New Generation	*
Galactic Trooper	Romik	6
Galaxians	Quicksilva	6
Galaxy Jailbreak	Romik	6
Games 2	JRS	*
Games Tape 1	J K Greye	*
Games Tape 2	J K Greye	*
Games Tape 3	J K Greye	*
Gamestape 1	Fawkes Computing	7
Glooper	Quicksilva	7
Gloops	Quicksilva	5
Gobbleman	Artic	*
Gobbler	Software Farm	*
Grand Prix	dk'tronics	*
Gulp 2	Campbell	*
Hang Glider	S Electronics	6
Hickstead	CCS	5
High-resolution Invaders	Odyssey	*
Invaders	Abersoft	*
Invaders	Bug-Byte	5
Invaders	Odyssey	8
Invaders	Selec	7
Invaders	Silversoft	*
Krazy Kong	PSS	*
Maze Death Race	PSS	*
Maze Man	Abersoft	*

Mazogs	Bug-Byte	8
Micro Mouse	Lothlorien	7
Namtar Raiders	Artic	8
Night Gunner	Digital Integration	5
Puckman	Hewson	5
Sabotage	Sinclair	6
Six Games	A Stubbs	5
Space Raiders	Sinclair	*
Space Trek	JRS	*
3D Monster Maze	New Generation	*
Three Games Cassette	McGraw Hill	8
Zuckman	DJL	8
ZX Invasion Force	Artic	*
ZX Panic	Selec	4
ZX-81 1K Games Pack	Crystal	9
ZX-81 Pocket Book	Phipps	*

Business

Accounts (Limited company)	Hestacrest	*
Accounts (Sole Trader)	Hestacrest	*
Business Bank Account	Transform	*
Critical Path Analysis	Hilderbay	6
Draft	Myrmidon	*
Mailing List	Hestacrest	*
Payroll	Hilderbay	*
Payroll	Soft Tech	*
Payroll	Soft Tech	*
Personal Banking System	V&H Computing	*
Purchase Ledger	Hilton	*
Sales Day Book	Hestacrest	*
Sales Ledger	Transform	*
Text	Hestacrest	8

Education

Calpac 1-2	Calpac	*
Four Rules of Number	Micro Master	*
Integration	University	8
Intermediate English 1-2	Rose	*
Intermediate Maths 1-2	Rose	*
Language Devel. Series	Glasson	*
Language Devel. Series	Micro Master	*
Linear Programming	University	7
Matrix Operations	University	*
O Level Chemistry	Calpac	*
O Level French Revision	Rose	*
O Level Maths Revision	Rose	*
Polynomials	University	*
Primary Arithmetic	Rose	*
Regression	University	*
Self-teach Program	Anvil	*

Language

Forth	Sinclair	7
ZX Forth	Artic	8

Practical

Ephemeris	Bridge	7
Football Pools	Hartland	*
Poolster	Naigram	3

Puzzle

Nowotnik Puzzle	Phipps	*
Word Fit	Ram Writer	6

Simulation

Fighter Pilot	Digital Integration	5
Flight Simulation	Sinclair	6
Pilot	Hewson	7
Print Shop	CCS	*

Strategy

Airline	CCS	7
Auto Chef	CCS	7
Battleships	JRS	7
Conflict	Martech	7
Cyborg Wars	Stratagem	7
Dallas	CCS	6
Dictator	Bug-Byte	7
Farmer	CCS	*
Football Manager	Addictive Games	7
Fort Apache	Contrast	8
Galaxy Conflict	Martech	8
Great Britain Ltd	Hessel	*
Ocean Trader	Quicksilva	7
Pioneer Trail	Quicksilva	7
Racehorse Trainer	G Barker	6
Racing League	Racing League Software	4

Traditional

Do Not Pass Go	Work Force	*
----------------	------------	---

Lynchmob	Bridge	6
Original Superchess	CP Software	*
Tai	PSS	*
Tenpin	Phipps	*
ZX 1K Chess	Artic	*
ZX Compendium	Mastervision	7
ZX-Chess I	Artic	*

Utility

Graphics	IPA	*
Graphics Toolkit	JRS	9
HI Resolution	CRL	6
Machine Code Test Tool	OCF	*
MCode	PSS	8
Programme Enhancement Package	R and R	*
Renumber Delete	Work Force	*
Trace	Texgate	7
ZX Compiler	Silversoft	*
ZX Screenkit	Picturesque	*
ZX-81 Remload	Artic	*
ZX-Bug	Microsphere	*
ZX-sideprint	Bug-Byte	*
ZXAS	Bug-Byte	*
ZXDB	Bug-Byte	*

SPECTRUM 16K

Adventure

Android One	Vortex	*
Escape	New Generation	8
Mines of Saturn/Return to Earth	Mikro-Gen	*
Moria	Severn	*
Planet of Death	Artic	6
Secret Valley	New Soft	*
The Great Western	New Soft	*
Time Bandits	New Soft	*

Arcade

Aquarius	Bug-Byte	6
Arcadia	Beau Jolly	5
Arcadian	J K Greye	6
Assassin	Spectrasoft	*
Avenger	Abacus	5
Baron	Temptation	*
Base Invaders	Imagination	*
Black Hole	Quest	6
Blind Alley	Sunshine	7
Bug Blaster	Crystal	*
Cassette A	Orwin	*
Caterpillar	CDS	*
Cavern Fighter	Bug-Byte	4
Centi-Bug	dk'tronics	*
Children's Compendium	Dymond	*
City Defence	Mikro-Gen	*
Colour Clash	Romik	4
Cookie	Ultimate	*
Cosmic Guerilla	Crystal	8
Crazy Cranes	Voyager	5
Creepy Crawler	Mikro-Gen	*
Crevasse and Hotfoot	Microsphere	7
Cruising	Sunshine	4
Cyber Rats	Silversoft	7
Death Chase	Micromega	6
Demolition	Comp. Rentals	*
Destroyer	Winters	*
Di-lithium Lift	Hewson	3
Digger Dan	Ocean	*
Doombugs	Work Force	8
Dymonoids	Dymond	*
Earth Defence	Artic	5
Ed-On	Add-On	4
Eskimo Eddie	Ocean	*
Family Games Pack	Hornby	5
Fireflash	Abacus	7
Froggy	DJL	7
Fruit Machine	dk'tronics	6
Galactians	dk'tronics	*
Galactic Trooper	Romik	3
Galactic Warriors	Abacus	6
Galaxians	Artic	6
Ghost Hunt	PSS	*
Gnasher	Mastertronic	4
Gobble-a-Ghost	CDS	*
Gobbleman	Artic	*
Godzilla and Martians	Temptation	7

Software Directory



Ground Attack	Silversoft	6
Gulpman	Campbell	*
Haunted Hedges	Micromega	6
Hopper	PSS	6
Horace and the Spiders	Sinclair	8
Horace Goes Skiing	Sinclair	8
Hungry Horace	Sinclair	8
Invasion Force	Artic	8
It's the Wooluf	Crystal	5
Jet Pac	Ultimate	8
Labyrinth	Axis	4
Leap Frog	CDS	*
Light Cycle	PSS	*
Luna Crabs	Micromega	7
Magic Meemies	CDS	2
Maze Chase	Hewson	8
Meteor Storm	Quicksilva	7
Meteoroids	dk'tronics	7
Meteoroids	Softtek	6
Micro Mouse	Lothlorien	6
Millypede	Add-on	7
Mined Out	Quicksilva	7
Moon Buggy	Visions	6
Mr Wong's Loopy Laundry	Artic	6
Muncher	Silversoft	*
Muncher	Silversoft	6
Nanas	Mikro-Gen	*
Orbiter	Silversoft	7
Ostron	Softtek	*
Pengy	Micromania	4
Pitman Seven	Visions	*
Planetoids	Sinclair	8
Proteus	Abacus	5
Rapedes	Visions	*
Repulsar	Softtek	8
Rider	Virgin	6
Road Toad	dk'tronics	7
Robot Panic	Soft Mill	1
Sam Spade	Silversoft	5
Santa	Artic	*
Sentinel	Abacus	7
Shark Attack	Romik	5
Sheer Panic	Visions	7
Slippery Sid	Silversoft	8
Space Fighter	Winters	*
Space Intruders	Quicksilva	7
Space Lanes	Cathedral	4
Space Raiders	Sinclair	6
Space Zombies	Mikro-Gen	*
Spec. Frogs/ Showdown	Artic	*
Spec. Gobbleman	Artic	*
Spec. Invaders	Artic	*
Spec. Invasion Force	Artic	7
Spec. Scramble	Work Force	7
Spectipede	Mastertronic	5
Spectral Invaders	Bug-Byte	8
Spectral Panic	Hewson	5
Spectres	Bug-Byte	6
Strike Four	Spectresoft	4
Styx	Bug-Byte	7
Sub	Romik	*
Tank Battle	dk'tronics	8
3D Monster Chase	Romik	3

Train Game	Microsphere	9
Trom	dk'tronics	4
Vortex	JK Greye	5
Winged Warlord	CDS	6
Wizard's Warriors	Abersoft	*
Yomp	Virgin	7

Business

Finance Manager	OCP	9
Home Computer Pack	SD Micro	*
Masterfile 16	Campbell	*
Matcalc	Work Force	*
Micropen	Contrast	*
Vu-Calc	Sinclair	7
Vu-File	Sinclair	8

Education

Alphabet	Widget	*
Alphabet Games	Sinclair	*
Apostrophe	Sinclair	7
Ballooning	Heinemann	*
Calpac 1-3	Calpac	*
Car Journey	Heinemann	*
Cargo	Sinclair	*
Chess Tutor	Artic	7
Counting	Starter Soft	*
Counting	Widget	*
Education One	Lerm	*
Educational	Startersoft	*
Firework Music	Soft Cottage	8
First Numbers	Collins	*
40 Education Games	Granada	*
Four Rules of Number	Micro Master	*
French Voc Test	Tutorial	*
Hidden Letters	Poppy	*
Hot Dot Spotter	Longman	*
Integration	University	8
Intermediate English 1-2	Rose	*
Intermediate Maths 1-2	Rose	*
Know Your Tables	Collins	*
Language Devel. Series	Glasson	*
Language Devel. Series	Micro Master	*
Learn Basic	Logic 3	*
Learning Read 1	Poppy	*
Linear Programming	University	*
Marks Book	Lerm	*
Maths Invaders	Stell	*
Maths Tutor	AD Software	*
Matrix Operations	University	7
Money	Poppy	*
Night Sky	Bridge	*
O Level Maths	Homestudy	*
O Level Physics	Homestudy	*
Paddington's Shopping Mix-up	Collins	*
Pathfinder	Widget	*
Polynomials	University	8
Pre/early school cassettes	Essex	*
Primary Arithmetic	Rose	*
Punctuation Pete	Heinemann	8
Regression	University	8
Self-teach Program	Anvil	*
Shape Sorter	Widget	*
Special Agent	Heinemann	*
Spellbin	Startersoft	*
Use and Learn	Microl	8

Language

Beta Basic	Betasoft	9
------------	----------	---

Practical

Biorythms	Spectrasoft	*
Countries of the World	Hewson	8
Cycle Planner	Medidata	4
Map of the UK	Kuma	7
Shopping List	SD	4
Spectasort	Spectadraw	*

Puzzle

Flippit	Sinclair	7
Hanoi King	Contrast	7
Lojix	Virgin	3
Nowotnik Puzzle	Phipps	7
Quazar	Rose	4

Simulation

Air Traffic Controller	Hewson	6
Airliner	Protek	6
Circus	R & R	7
Classic Adventure	Virgin	7

Software Directory

Golf	Virgin	8
NightFlite	Hewson	5
Print Shop	CCS	8
Pro-Golf	Hornby	7

Strategy

Auto Chef	CCS	*
Big Match Soccer	Winters	*
Dallas	CCS	6
Dictator	dk'tronics	*
Farmer	CCS	*
Football	Winters	*
Heathrow	Hewson	8
Las Vegas	Temptation	*
Quincy	Severn	*

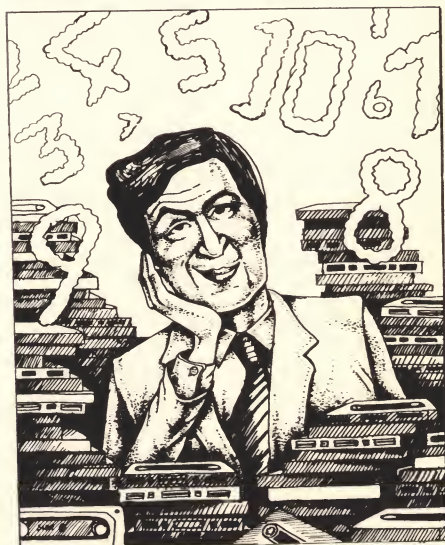
Traditional

Backgammon	Hewson	*
Bridge Tutor	CP Software	8
Bridgemaster	Bridgemaster	9
Challenge	Temptation	*
Gambling Tape	Dymond	*
Las Vegas	Temptation	5
Odds-on	RSD	*
Othello	CP Software	*
Pinball	Winters	*
Pool	Bug-Byte	7
Reversi	Sinclair	8
Roulette	Newsoft	*
Solo Whist	Video Soft.	*
Spec. Microchess	Artic	*
Super Play I	Video	*
Tennis	Winters	*

Utility

Aspect	Bug-Byte	5
Audio Sonics	Work Force	4
Auto Sonics	Buttercraft	5
Basic Utilities	Jaysoft	6
Character Generator	Spectrasoft	*
Dietron	Custom	6
Disassembler	dk'tronics	7
Display	Work Force	*
Editor/Assembler	Picturesque	8
Extended Basic	CP Software	7
FP Compiler	Softek	9
Friendly Face	Monitor	4
Keysounder	S and G	*
Letterfont	Allanson	8
Machine Code Test Tool	OCP	7
Master Toolkit	OCP	7
MCoder	PSS	8
Micropen	Contrast	*
Print Utilities	Sinclair	7
Programmer's Dream	Work Force	8
Renumber Delete	Work Force	*
Slow Loader	ELR	9
Sound FX	dk'tronics	6
Spec. Bug	Artic	*
Spec. Editor/ Assembler	Picturesque	8
Spec. Monitor	Picturesque	7
Spectrum Super Toolkit	Nectarine	8
Spectsound	PDQ	2
Supercode	CP	8
Taswide-64	Tasman	*
Trace	Texgate	8
TT-S	Timedata	8
ZX Spectrum Assembler	McGraw Hill	4
ZXED	dk'tronics	*

Colditz	Phipps	8	Carnival	Eclipse	5	Rider	Virgin	6
Colossal Caves	CP Software	8	Carpet Capers	Terminal	7	River Rescue	Creative Sparks	3
Cry Wolf!	Add-on	*	Cavelon	Ocean	7	Robot Riot	Silversoft	6
Demon Lord	MCE	5	Centipoid Plus 3	Orwin	5	Rommels Revenge	Crystal	8
Detective	Archie	7	Chequered Flag	Sinclair	8	Sabre Wulf	Ultimate	8
Devils of Deep	Shepherd	7	Chuckie Egg	A & F	8	Scuba Dive	Durrell	9
Diamond Quest	CCS	4	Chuckman	CCI/Add-on	8	Security Shelter	Add-on	*
Diamond Trail	Gilsoft	7	Close-In	Pulsonic	4	Skull	Games Machine	5
Dragonsbane	Quicksilva	6	Codename Mat	Micromega	9	Sorcery	Virgin	7
Dungeon Master	Crystal Comp.	8	Corridors of Genon	New Generation	9	Space Station Zebra	Beyond	6
Dungeons of Doom	Temptation	3	Cruise Attack	Mikro-Gen	*	Spectron	Virgin	6
Espionage Island	Artic	5	Cyber Zone	Crystal	7	Spellbound	Beyond	7
Everest Ascent	Shepherd	6	Death Chess 5000	Artic	7	Splat	Incentive	8
Fantasia Diamond	Hewson	7	Defenda	Interstella	2	Stop the Express	Sinclair	8
Golden Apple	Artic	7	Defendar	Mikro-Gen	*	Submarine Strike	Pulsonic	6
Gorgon	Phipps	*	Defusion	Incentive	6	Tank Trax	Mastertronic	3
Halls of Things	Crystal Comp.	9	Defusion/Worms	K-Tel	3	The Guardian	PSS	6
Here comes the sun	Alligata	7	Demon	Microcosm	7	The Pyramid	Fantasy	7
Hobbit	Melbourne House	9	Demon Chase	Mansfield	4	The Snowman	Quicksilva	8
Hole	Add-on	*	Devil Rides In	Mastertronic	7	3D Bat Attack	Cheetahsoft	6
Horror Atoll	Add-on	*	Dimension Destroyers	Artic	8	3D Lunattack	Hewson	5
Inca Curse	Artic	6	Dinky Digger	Postern	4	3D Seiddab Attack	Add-on	1
Inferno	Shepherd	7	Dodge City	Phoenix	7	3D Star Wars	New Generation	9
Invincible Island	Shepherd	8	Dr Franky and the Monster	Virgin	6	3D Tunnel	Quicksilva	7
Island	Crystal	4	Driller Tanks	Sinclair	3	Time Gate	Quicksilva	8
Island	Virgin	7	Elektro Storm	PSS	*	Timebomb	Add-on	7
Jericho Road	Shards	7	Energy 30,000	Elm	3	Tobor	Vortex	8
Jungle Adventure	CCS	4	Eric and the Floaters	Sinclair	6	Tornado Low Level	Ultimate	*
Knight's Quest	Phipps	6	Eskimo Eddie	Ocean	4	Tranz Am	New Generation	8
Leopard Lord	Add-on	*	Frank N Stein	PSS	7	Trashman	Quicksilva	7
Lords of Midnight	Beyond	9	Fred	Quicksilva	6	Traxx	Software Projects	8
Lords of Time	Level Nine	8	Freez Beez	Silversoft	*	Tribble Trouble	dk'tronics	4
Lost Over Bermuda	Add-on	*	Froot Loop	NTD Software	5	Trom	Micromania	8
Mad Martha	Mikro-Gen	7	Galaxy Attack	Sunshine	4	Tutankhamun	Lothlorien	7
Mad Martha II	Mikro-Gen	*	Glug Glug	CRL	6	Two-Gun Turtle		
Mountains of Ket	Incentive	8	Harry Goes Home	Pulsonic	6	Warlock of Firetop		
Murder at Manor	Gemtime	7	Hickstead	CCS	4	Mountain	Penguin	*
Mysterious Fairground	Buffer Micro	6	High Noon	Work Force	9	Wheelie	Microsphere	7
Odyssey of Hope	Martech	6	House of Living Dead	Phipps	7	Worm Attack	Pulsonic	2
Oracle's Cave	Doric	8	Hunchback	Ocean	7	Worse Things Happen at Sea	Silversoft	7
Orb	Comp. Rentals	*	Invasion Body Snatch.	Crystal	9	Xadom	Quicksilva	4
Paradox	Runesoft	2	Jack and the Beanstalk	Thor	5	Zig-Zag	dk'tronics	7
Perseus and Andromeda	Channel 8	7	Jackpot	Comp. Rentals	*	Zipper Flipper	Sinclair	4
Pimania	Automata	7	Jet Set Willy	Software Projects	9			
Quest	Hewson	7	Killer Knight	Phipps	5	Business	Fulwood	*
Quetzalcoatl	Virgin	8	Knight Rider	Hewson	*	Account Management		
Roundsby Incident	Add-on	*	Kosmic Kanga	Micromania	5	System		
Satan's Pendulum	Minatronic	7	Krakatoa	Abbex	8	Accounts (Limited		
Sherlock	Melbourne House	9	Krazy Kong	PSS	*	Company)	Hestacrest	*
Ship of Doom	Artic	7	Laser Zone	Quicksilva	8	Accounts (Sole Trader)	Hestacrest	*
Solaris	Sofitel	*	Laserwar	Mikro-Gen	6	Address File	SD Micro	7
Spoof	Runesoft	7	Last Sunset Lattica	Archie	6	Address Manager	OCF	8
Snowball	Level 9	9	Lazatron	Contrast	*	Bank Account System	K Gouldstone	8
Superspy	Shepherd	5	Les Flies	PSS	6	Bank Verifier	SD Micro	*
Temple of Vran	Incentive	8	Loony Zoo	Phipps	6	Business Bank Account	Shepherd	7
Terror from The Deep	Add-on	*	Lunar Jetman	Ultimate	8	Cash Controller	Sinclair	7
Time Quest	Mikro-Gen	*	Manic Miner	Bug-Byte	9	Collector's Pack	Hilderbay	*
Titanic	R&R	5	Matrix	Salamander	*	Critical Path Analysis	Microl	7
Transylvanian Tower	Shepherd	7	Maze Death Race	PSS	8	Database	RAMTOP	*
Urban Upstart	Shepherd	*	Maziacs	dk'tronics	8	DIY Book-keeping	Heath Computing	7
Valhalla	Legend	7	Metagalactic Llamas	Salamander	6	Investment, Insurance,		
Vampire Village	Terminal	4	Mission Impossible	Silversoft	6	Information	Inform	5
Velnor's Lair	Quicksilva	8	Mission Omega	Pulsonic	7	Masterfile	Campbell	8
Volcanic Dungeon	Mastervision	6	Monkey Biznes	Artic	4			
Width of the World	Mosaic	4	Moon Alert	Ocean	5			
Wrath of Magma	Mastervision	8	Moons of Tantalus	Cornhill	7			
Ziggurat of Dread	Add-on	*	Mr Wimpey	Ocean	7			
			Mummy Mummy	Lothlorien	6			
			Munnery's Mergatroids	Abacus	6			
Archie				Visions	4			
Ad Astra	Gargoyle Games	8		Automata	6			
Adven. of a St Bernard	Mastertronic	7		Software Projects	5			
Alcatraz Harry	Mastertronic	3		Rabbit	4			
Android Two	Vortex	*		Mikro-Gen	*			
Ant Attack	Quicksilva	8		Beau Jolly	7			
Antics	Bug-Byte	8		Melbourne House	7			
Arena 3000	Microdeal	5		Automata	7			
Armageddon	Silversoft	3		Automata	6			
Atic Atac	Ultimate	8		Proffisoft	6			
Automania	Mikro-Gen	6		Postern	5			
Base Invaders	Work Force	6		Ultimate	7			
Bear Bover	Artic	8		Beyond	9			
Birds and Bees	Bug-Byte	8		Fantasy	8			
Blade Alley	PSS	6		Archie	*			
Blue Thunder	Foundry Systems	6		Bug-Byte	6			
Brain Damage	Silversoft	*		Gemini	7			
Bubble Buster	Sinclair	5		Comp. Rentals	*			
Bubble Trouble	Archie	7		Ocean	*			
Bugaboo	Quicksilva	*						
Butterfly	Pulsonic	4						
Caesar the Cat	Mirrorsoft	7						



Micropen	Contrast	5	Logo	Sinclair	9	1984	Incentive	7
Multi-File	ISP	6	Micro Prolog	Sinclair	8	Oligopoly	CCS	7
Omnicale	Microsphere	9	Pascal Compiler	Hi-Soft	*	Plunder	CCS	8
Payroll	Hilderbay	9	Snail Logo	CP	8	Red Weed	Lothlorien	7
Payroll	Transform	*	Spec. Forth	Abersoft	9	Reichswald	MW Gamesworld	3
Payroll	V&H Computing	*	Spec. Forth	CP	8	Scatterbrain	Manor	7
Personal Banking System	Hilton	*	ZX Forth	Artic	8	Sheepwalk	Virgin	7
Personal Financ.						Special Operations	Lothlorien	8
Management Syst.	Fulwood	*	Practical			Spectrealm	Runesoft	6
Sales Day Book	Transform	*	Beamscan	Beamscan	9	Star Trader	Bug-Byte	6
Small Business Accounts	Sinclair	*	Computer Cookbook	Bug Byte	5	Stonkers	Beau Jolly	9
Spectext	McGraw Hill	7	Cricketer Averages	Spartan CC	7	Super-League	Cross	6
Spreadsheet	Microl	7	Diet	dk'tronics	*	Tradewind	WDS	*
Stock Control	Hilderbay	*	Diet Master	Diet Master	6	War 70	CCS	8
Stock Control	Kemp	*	Dietician	Keysoft	4	Warlord	Lothlorien	*
Stock Control	Transform	*	Engine Diagnostic	Spectrasoft	*	Whodunnit	CCS	8
Tasword	Tasman	8	First Aid	Eastmead	4	Wilfred the Hairy	Microbyte	2
Tasword II	Tasman	9	Football Pools	Hartland	*			
Utility File	SD Micro	*	How Long have you got?	Eastmead	6	Traditional		
Word Processor	Microl	2	I Ching	Salamander	7	Arcturus	Visions	7
Word Processor	Quicksilva	2	I Ching	Sirius	*	Backgammon	CP Software	7
			Personal Reminder	SD Micro	*	Brag	T Lebon	7
			Spectadraw 2	McAlley	*	Bridge Master	Serim	7
			Star Gazer	CRL	8	Bridge Player	CP Software	8
			The Complete Guide to			Bullseye	Mastertronic	4
			Medicine	Eastmead	4	Derby Day	CRL	5
			Vega-Table	Vega	7	Do Not Pass Go	Work Force	6
			World Info	Winsoft	*	Double Dealer	MFM Software	5
						Draughts	CP Software	7
			Puzzle			Evolution	Microsphere	*
			Arcturus	Visions	7	Go To Jail	Automata	*
			Computaword	Work Force	2	Grid run/Pontoon	Archie	*
			Flix	Softricks	7	Mind Games	Oasis	7
			Hanoi King	Contrast	*	Original Superchess	CP Software	*
			Jumbly	dk'tronics	*	Pontoon	Contrast	3
			Mazecube	PAL	6	Rampoly	J Fletcher	*
			Stuart Henry's Pop Quiz	Bellflower	7	Roulette	Dymond	*
			3D Strategy	Quicksilva	8	Scrabble	Sinclair	8
						Snooker	Visions	*
			Simulation			Super Bridge	Buffer Micro	7
			Ashes	Pulsonic	7	Superchess II	CP Software	8
			Cricket Captain%Allanson			Superchess III	CP Software	9
			Fighter Pilot	Digital Integration	*	The Turk	OCF	8
			Flight Simulation	Sinclair	6	Voice Chess	Artic	9
			Full Throttle	Micromega	8	Yahtzi	Work Force	*
			Golf	dk'tronics	*	Yatze	CP Software	5
			Howzat	Wyvern	8	ZX Draughts	CP Software	7
			Inkos	Chalksoft	*	ZX Reversi	CP Software	7
			Match Point	Sinclair	9	ZX-Chess II	Artic	*
			New Birkdale	Hornby	8			
			Olympics	CRL	6	Utility		
			Royal Birkdale	Ocean	8	Allsort S-1	A Firminger	7
			Strike Attack	Micromart	4	Assembler	Artic	6
			Super Soccer	Winters	4	Beyond Basic	Sinclair	5
			Test Match	Comp. Rentals	6	Building Price	J Redman	6
			The Forest	Phipps	*	Cartoon Animation	Fowler	*
			The Forest	Phipps	7	Character Generator	ISP	8
			Troon	Hornby	8	Compiler	Softek	6
			United	CCS	4	Composer	Contrast	2
			World Cup Football	Artic	7	DLAN	Campbell	*
						FP Compiler	Softek	9
			Strategy			Games Designer	Quicksilva	7
			Airline	CCS	7	HURG	Melbourne House	8
			Angler	Virgin	6	Keyword Extension	Timedata	8
			Apocalypse	Red Shift	*	Linked Software:		
			Battle 1917	CCS	8	Information Handling	McGraw Hill	3
			Battle of Britain	Microgame	*	List File	SD Micro	*
			Brewery	CCS	8	Make Music	Buffer Micro	8
			British Lowland	CCS	*	Melbourne Draw	Melbourne House	7
			Caribbean Trader	E. Midland	7	Monitor/Diss.	Sinclair	8
			Conflict	Martech	*	Music Maker	Bellflower	3
			Conquest	Cheetahsoft	6	Print & Plotter	ISP	8
			Dix Mille	CCS	5	Print Utilities	Sinclair	7
			Fall of Rome	ASP	9	Quill	Gilsoft	9
			Football Manager	Addictive Games	6	Screen Machine	ISP	7
			Galaxy Conflict	Martech	7	Softalk 1-2	CP Software	8
			Galaxy Conflict	Martech	8	Spec. Assembler	Artic	*
			Gangsters	CCS	8	Spec. Compiler	Softek	7
			Gatecrasher	Quicksilva	7	Spectre Mac/Mon	Oasis	8
			General Election	Spectrosim	7	Spectrum	Shiva	4
			Golf	Bug-Byte	1	Spectrum Extended Basic	CP Software	7
			Great Britain Ltd	Hessel	7	Spectrum Monitor	Picturesque	7
			Hunter Killer	Protek	9	Spectrum Sprites	ISP	8
			It's Only Rock 'n' Roll	K-Tel	5	Spectrum Super Toolkit	Nectarine	8
			Johnny Reb	Lothlorien	*	Supercode II	CP Software	8
			King Arthur	E. Midland	*	The Complete Machine Code		
			Millionaire	Trans Express	7	Tutor	New Generation	8
			Mugsy	Melbourne House	8	Romantic Robot	Sinclair	8
			New Venture	Falcon	4	Zeus Assembler	Sinclair	9
Language								
Forth	Melbourne House	9						
Forth	Sinclair	7						
48/80 Forth	E London							

Software Publishers

A & F Software, Unit 8, Canalside Industrial Estate, Woodbine Street East, Rochdale, Lancashire OL16 5LB
Abacus Software, 21 Union Street, Ramsbottom, Nr Bury, Lancashire
Abbex, 20 Ashley Close, Manor Hall Drive, London NW4
Abersoft, 7 Maesfallen, Bow Street, Aberystwyth, Wales
Add-on Electronics, Units 2,3 & 4, Shire Hill Industrial Estate, Saffron Walden, Essex CB11 3AQ
Addictive Games, 7a Richmond Hill, Bournemouth BH2 6HE
ADS, 8 Bronchurch Street, Portsmouth, Hampshire PO4 8RY
Allanson Computing, 77 Chorley Road, Adlington, Chorley, Lancashire PR6 9LH
Alligata Software, 1 Orange Street, Sheffield S1 4DW
APS, 1 Golden Square, London W1
Arcaide Software, Technology House, 32 Chislehurst Road, Orpington, Kent BR6 0DG
Arnold Wheaton, Parkside Lane, Dewsbury Road, Leeds LS11 5TD
Artic Computing, Main Street, Brandesburton, Driffield YO25 8RG
Automata UK, 27 Highland Road, Portsmouth, Hampshire PO4 9DA
Axis, 71 Brookfield Avenue, Loughborough, Leicestershire LE11 3LN
Beau-Jolly, 19A New Broadway, Ealing, London W5
Bellflower Software, 6 Rosewood Avenue, Greenford, Middlesex UB6 7QP
Betasoft, 92 Oxford Road, Moseley, Birmingham B13 9SQ
Beyond Software, 8 Herbal Hill, London EC1
Bridge Software, 36 Fernwood, Marple Bridge, Stockport, Cheshire SK6 5BE
Bridgemaster, Sandymouth, Beeches Road, Farnham Common, Buckinghamshire SL2 3PS
Buffer Micro, 310 Streatham High Road, London SW16
Bug-Byte, Mulberry House, Canning Place, Liverpool L1 8JB
Calpac Computer Software, 108 Hermitage Woods Crescent, St Johns, Woking, Surrey
Campbell Systems, 57 Trap's Hill, Loughton, Essex IG10 1TD
CCS, 14 Langton Way, Blackheath, London SE3 7TL
CDS Micro Systems, Silver House, Silver Street, Doncaster, South Yorkshire DN1 1HL
Chalksoft, 37 Willowslea Road, Northwick, Worcester
Cheetahsoft, 24 Ray Street, London EC1R 3DJ
Clever Clogs, Argus Press Software Group, 1 Golden Square, London W1R 3AB
Collins, 18/20 Stephenson Way, North Gower Street, London NW1 2DX
Computatutor, 3 Thalia Close, Greenwich, London SE10 9NA
Contrast Software, Warren Road, Liss, Hampshire GU33 7DD
Cornhill Software, 2 Penrith Way, Aylesbury, Buckinghamshire HP21 7JZ
CP Software, 1 Glebe Road, Uxbridge, Middlesex UB8 2RD
Creative Sparks, Thomson House, 296 Farnborough Road, Farnborough, Hampshire
Cross Software, 36 Langford Crescent, Barnet, Hertfordshire EN4 9EH
Crystal Computing, 2 Ashton Way, East Herrington, Sunderland SR3 3RX
Digital Integration, Watchmoor Trade Centre, Watchmoor Road, Camberley, Surrey GU15 3AJ
dk'tronics, Unit 6, Shire Hill Ind Est, Saffron Walden, Essex CB11 3A
Doric Computer Services, 3 The Oasis, Glenfield, Leicester LE3 8QS
Durrell Software, Castle Lodge, Castle Green, Taunton TA1 4AB
Dymond Software, 22 Hospital Road, Annan, Dumfriesshire DG12 5HP
East London Robotics, St Nicholas House, The Mount, Guildford, Surrey GU2 5HN
Eastmead, Eastmead House, Lion Way, Camberley, Surrey GU16 5EZ
Ebury Press, Humphrey Bull & Barker, 62 Dean Street, London W1V 5HG
Elm Computers, 59 Bateman Road, East Leake, Loughborough, Leicestershire LE12 6NN
Fantasy Software, Fauconberg Lodge, 27a St Georges Road, Cheltenham
Fawkes Computing, 41 Wolfridge Ride, Alveston, Bristol BS12 2RA
Felix Software, 19 Leighton Avenue, Pinner HA5 3BW
Fulwood, 20 Templestowe Hill, Whitkirk, Leeds LS15 7EJ
Gargoyle Games, 4 North Western Arcade, Birmingham BS 5LH
Gemini Software, 18a Littleham Road, Exmouth, Devon EX8 2QG

Gemtime Software Division, 16 Ben Ledi Road, Kirkcaldy, Fife KY2 5RP
Gilsoft, 30 Hawthorn Road, Barry, South Glamorgan, South Wales
Gouldstone, 45 Burleigh Avenue, Wallington, Surrey SM6 7UG
Granada Publishing, 8 Grafton Street, London W1X 3LA
Griffin & George, Frederick Street, Birmingham B1 3HT
Harlequin Software, 43 Osprey Park, Thornbury, Bristol BS12 1LY
Hartland Software, 32 Ivor Place, London NW1 6DA
Heath Computing, 7 The Meadows, Flackwell Heath, Buckinghamshire HP10 9LX
Hessel, 15 Lythan Court, Cadwell Crescent, Sunningdale, Berkshire
Hestacrest, PO Box 19, Leighton Buzzard, Bedfordshire LU7 0DG
Hewson Consultants, 56B Milton Trading Estate, Milton, Abingdon
Hilderbay, 8/10 Parkway, Regents Park, London NW1 7AA
Hilton Computer Services, 14 Avalon Road, Orpington, Kent
Hisoft, 180 High Street, Dunstable, Bedfordshire LU6 1AT
Homestudy Ltd, Treleigh Woods Farm, Treleigh, Redruth, Cornwall TR16 4AW
Hornby Software, 21 Penfold Hill, Leeds LS15 0PW
Image Systems, 34 Lynwood Drive, Worcester Park, Surrey KT4 7AB
Incentive, 54 London Street, Reading, Berkshire RG1 4SQ
Inform Software, 3 Treesdale Close, Birkdale, Southport PR8 2EL
Interstella Software, 82 New Forest Drive, Brockenhurst, Kent
ISP Marketing Ltd, Crown Hill, 38B High Street, Godalming, Surrey GU7 1DZ
JK Greye Software, 16 Park Street, Bath, Avon BA1 2TE
JRS Software, 19 Wayside Avenue, Worthing, Sussex BN13 3JH
K-Tel International (UK), 60 Western Avenue, London W3 0TU
Kemp, 43 Muswell Hill, London N10 3PN
Keysoft, 6 Bruce Grove, Tottenham, London N17
Kosmos Software, 1 Pilgrims Close, Harlington, Dunstable, Bedfordshire LU5 6LX
Kuma Computers, 12 Horseshoe Park, Pangbourne, Berkshire RG8 7JW
Legend, PO Box 435, London E4 7LX
Lerm, 10 Brunswick Gardens, Corby, Northamptonshire
Level Nine, 229 Hugenden Road, High Wycombe, Buckinghamshire HP13 5PG
Logic 3, Unit 18, Wye Ind Est, London Road, High Wycombe, Buckinghamshire
Longman, Longman Group, Longman House, Harlow, Essex CM20 2JE
Lothlorien, 56a Park Lane, Poynton, Stockport, Cheshire SK12 1RE
Manor Software, 24 Manor Gardens, London SW20
Martech Games, 9 Billingham Road, Eastbourne, East Sussex BN20 8LY
Mastertronic, (as for Mastervision)
Mastervision, Park Lorne, 111 Park Road, London NW8 7SL
McAlley, 1 Cowleaze, Chinnor, Oxfordshire OX9 4TD
McGraw Hill, Shoppen Hangers Road, Maidenhead, Berkshire
Medidata, PO Box 26, London NW9 9BW
Melbourne House, Castle Yard House, Castle Yard, Richmond TW10 6TF
Micro Dealer UK, Unit 6, Marlborough Road Trading Estate, Lattimore Road, St Albans, Hertfordshire
Microbyte, 19 Worcester Close, Lichfield, Staffordshire
Microcosm, 68 The Glade, Clayhall, Ilford
Micromania, 14 Lower Hill Road, Epsom, Surrey KT19 8LT
Micromega, 230/236 Lavender Hill, London SW11
Microsphere, 72 Roseberry Road, London N10 2LA
Mikro-Gen, 44 The Broadway, Bracknell, Berkshire RG12 1AG
Minatron Computing, 34 Pinewood Close, Westbury on Trym, Bristol BS9 4AJ
Mirrorsoft, PO Box 50, Bromley, Kent BR2 9TT
Monitor Software, PO Box 442, London NW7 2JF
Mosaic, 187 Upper Street, London N1 1RQ
MW Gamesworld, 12 Lawnswood Avenue, Chasetown, Walsall WS7 8YD
Myrmidon Software, PO Box 2, Tadworth, Surrey KT20 7LU
Naigram Software, c/o Soho Synth House, 18A Soho Square, London W1V 5FB
Nectarine, 837 Yeovil Road, Slough SL1 4JH
Newsoft, 12 White Broom Road, Hemel Hempstead, Hertfordshire
NTD Software, 19 Radipole Road, London SW6

Oasis Software, 9a Alexandra Parade, Weston-Super-Mare, Avon BS23 1QT
Ocean Software, 6 Central Street, Manchester M2 5NS
OCP, 4A High Street, Chalfont St Peter, Buckinghamshire SL9 9QB
Orwin Software, 26 Brownlow Road, Willesden, London NW10 9QL
Penguin, 536 King's Road, London SW10
Phipps Associates, 172 Kingston Road, Ewell, Surrey

Phoenix Publishing, 14 Vernon Road, Bushey, Hertfordshire WD2 2JL
Picturesque, 6 Corkscrew Hill, West Wickham, Kent BR4 9BB
Poppysoft, The Close, Common Road, Headley, Newbury, Berkshire
Print 'n' Plotter Products, 19 Borough High Street, London SE1 9SE
Protek Computing, 1a Young Square, Brucefield Ind Park, Livingston, West Lothian
PSS, 452 Stoney Stanton Road, Coventry CV6 JDG
Pulsonic, Warwick Distribution Ltd, 3 Standard Road, Park Royal, London NW10 6EX
Quicksilver, Palmerstone Park House, Southampton, Hampshire SO1 1LL
R and R, 34 Burton Road, Gloucester GL4 0LE
RAM Writer, 3 Vumba House, 2 Cedar Gardens, Sutton, Surrey
Ramtop Services, 5 Rue D'Artois, 75008, Paris, France
Red Shift, 12c Manor Road, Stoke Newington, London N16 5SA
Romantic Robot, 113 Melrose Avenue, London NW2 4LX
Romik Software, 272 Argyll Avenue, Slough SL1 4HE
Rose Software, 148 Widney Lane, Solihull, West Midlands
Runesoft, Charnwood House, Crossgate Drive, Nottingham NG2 TLW
Salamander Software, 27 Ditchling Rise, Brighton, East Sussex BN1 4QL
Scisoft, 5 Minster Gardens, Newthorpe, Eastwood, Nottingham NG16 2AT
SCR Adventures, 190 Shelbourne Road, Tottenham, London
SD Microsystems, 9 Cadwell Court, Hitchin, Hertfordshire SG4 0AQ
Selec Software, 37 Councillor Lane, Cheadle, Cheshire
Serim Software, Freepost, Dept SU7, PO Box 163, Slough, Berkshire SL2 3YY
Shards, Suite G, Roycraft House, 15 Linton Road, Barking, Essex
Shepherd Software, Elm House, 23-25 Elmshott Lane, Chippenham, Slough, Berkshire
Silversoft, London House, 271/273 King Street, London W6 9LZ
Sinclair Research, 6 Kings Parade, Cambridge
Soft Cottage, 19 Westfield Drive, Loughborough, Leicestershire LE11 3QJ
Soft Tech, 31 Lampits, Hoddesdon, Hertfordshire
Softek International, 12/13 Henrietta Street, Covent Garden, London WC2E 8LH

Softel, 5 Durward Drive, Glenrothes, Fife KY6 2LB
Software Farm, Craig Farm, Botany Bay, Tintern, Gwent
Software Projects, Bear Brand Complex, Allerton Road, Woolton, Liverpool, Merseyside L25 7SE
Spartan CC, 29 Feltham Avenue, East Moseley, Surrey KT8 9BJ
Spectadraw, 1 Cowleaze, Chinnor, Oxfordshire OX9 4TD
Spectrasoft, Capital House, Market Place, London W3 6AL
Startersoft, 32 Parkfields, Chippenham, Wiltshire
Stell Software, 36 Limefield Avenue, Whalley, Lancashire BB6 9RJ
Stratagem Cybernetics, 286 Corbin Place, 2E, Brooklyn, New York 11235
Sulis Software, 4 Church Street, Abbey Green, Bath BA1 1FP
Sunshine Books, 12/13 Newport Street, London WC2
Sussex Publication, Townsend Poulshot, Devizes, Wiltshire SN10 1SD
Tasman Software, 17 Hartley Crescent, Leeds LS6 2LL
Temptation Software, 27 Cinque Ports Street, Rye, East Sussex
Terminal Software, Derby House, Derby Street, Bury BL9 0NW
Texgate, 14 Brook Lane, Corfe Mullen, Wimbourne, Dorset
Think Tank, 35 Wellington Road, Wimbledon Park, London SW19
Timedata, 16 Hemmells High Road, Laindon, Basildon, Essex SS15 6ED
Transform, 41 Keats House, Porchester Mead, Beckenham, Kent
Tutorial Software, 'Vilands', Glasllwch Lane, Newport, Gwent NP1 3PS
Ultimate Play the Game, The Green, Ashby de la Zouche, Leicestershire LE6 5JU
Unicorn Micro Systems, 312 Charminster Road, Bournemouth BH8 9RT
University Software, 29 St Peters Street, London N1
Vega Space Systems, 28 Watford Road, St Albans AL1 2AJ
Virgin Games, 2-4 Vernon Yard, London W11
Visions, 1 Feldgate Mews, Studlands Street, London W6
Vortex Software, 280 Brooklands Road, Brooklands, Manchester M23 9HD
Widget Software, 48 Durham Road, London N2 9DT
Wilcox Software, Station Road, Walsall WS7 6JZ
Winters, 24 Swannington Close, Cantley, Doncaster, South Yorkshire
Woosoft, 5 Andrews Close, Robertsbridge, Sussex TN32 5PB
Workforce, 140 Wilsden Avenue, Luton, Bedfordshire
Wyvern Software, 2 Princes Building, George Street, Bath BA1 2ED

HEARING IS BELIEVING Tele Sound SPECTRUM BEEP BOOSTER

TELESOUND is ready built and tested so that you can get the full sound of your Spectrum through any unmodified television set, black and white or colour. The unit is easy to fit in minutes without any previous experience as full fitting instructions are provided.

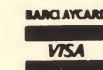
TELESOUND has been designed using the very latest ultra miniature components so it's size is an incredible 2 x 1.3 x 1mm. and fits neatly inside the Spectrum. Three easy push-on connections are made to the Spectrum. The connections are clearly shown in a diagram that comes with full fitting instructions.

TELESOUND FEATURES

- SOUND AND VISION TOGETHER FOR ADDED REALISM
- BEEP VOLUME CONTROLLABLE FROM A WHISPER TO A ROAR
- KEYBOARD CLICK CLEARLY HEARD TO ASSIST PROGRAM ENTRY
- PROGRAMS CAN BE HEARD WHEN BEING LOADED
- ADDITIONAL AMPLIFIER NOT NECESSARY
- SEPARATE POWER SUPPLY NOT REQUIRED
- USES NO MEMORY SO WORKS WITH ALL PROGRAMS
- COMPATIBLE WITH ALL OTHER ADD-ONS INCLUDING MICRODRIVE

TELESOUND comes complete with easy to follow fitting instructions and costs £9.95 inclusive of post, packing etc. Please state your computer when ordering.

COMPUSOUND



DEPT: SUA 32-33 LANGLEY CLOSE, REDDITCH, WORCS. B98 0ET. TELEPHONE: (0527) 21429 (21439 Answerphone).

NOW 3RD GREAT YEAR!
HIRE
 SPECTRUM and ZX81
PROGRAM TAPES

HUGE RANGE!
 ADVENTURE GAMES
 ARCADE and SIMULATION GAMES
 BUSINESS and PRACTICAL PROGRAMS
 UTILITIES
 M/C COMPILERS
 EDUCATIONAL
 FROM OVER 60 TOP SUPPLIERS

FREE
 ILLUSTRATED QUARTERLY MAGAZINE WITH TIPS, NEWS, REVIEWS, DISCOUNT OFFERS
LIFE MEMBERSHIP
 £5.00
 TAPE HIRE FROM 50P INCLUDING VAT
OVER 3,000 DELIGHTED MEMBERS!

The first true software library to be set up in the world—and still the largest. ■ Unrivalled tape stock, as you'd expect from our head start. ■ Run by a 16-bit computer (2 million bytes of program and data on disk!) with the help of six less able but more friendly mere humans! ■ Often imitated (sometimes almost to the letter—flattery indeed!) but never equalled. ■ Friendly, helpful service—at the end of a phone if needed. ■ Free membership of our Discount Club: pounds off peripherals and top-of-the-chart tapes. ■ Any loading problems (rare!) automatically credited. ■ Shop now open (next to Post Office) for rental and discount sales of tapes and peripherals. ■ Branches throughout the world. ■ No complicated schemes, no commitment to quantity—rent what and when you like. ■ Printed, illustrated magazine "Computerchat" packs in more than any mere typed newsletter. ■ Keenest-ever rates. ■ ACCESS holders can join by telephone on 0730 892732. ■ NO RIVAL can offer all this....

SPECIAL OFFER!
 YOUR FIRST TWO TAPES FREE

IF YOU USE THIS COUPON!
 or send £1 for magazine & details.
 Props: ALEC FRY ARPS, FRSA and ERNA FRY, BA

To: SOFTWARE LIBRARY, Farnham Road, West Liss, Hants GU33 6JU.
 NAME.....Machine.....

ADDRESS.....

THE ORIGINAL SINCLAIR LIBRARY!
 Special offer from this issue: **LIFE MEMBERSHIP £5** (normally £9.50). Overseas (Europe only) £10, or join local branch in W. Germany, France, Holland, Belgium, Scandinavia, South Africa, Ireland—send Int. Reply Coupon and we'll forward your enquiry. SA

Sinclair spares and repairs

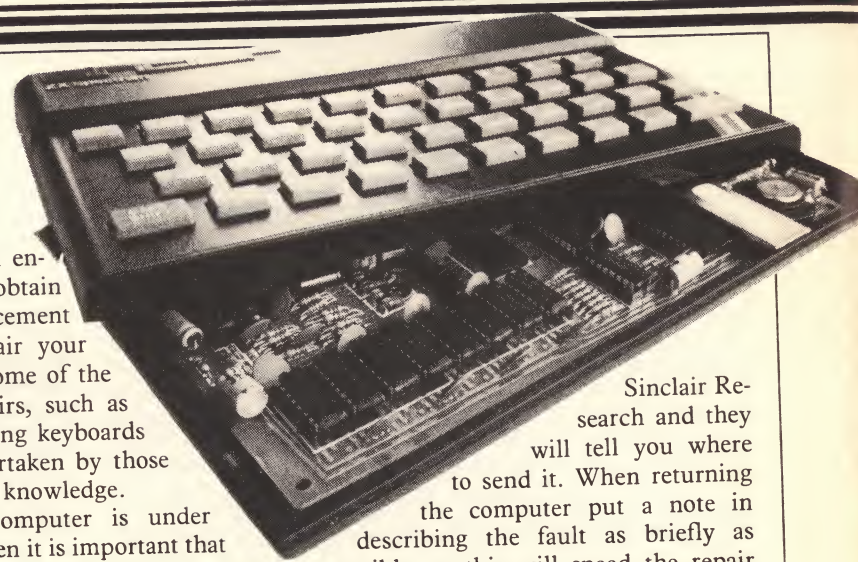
IF, LIKE MANY other Spectrum owners, you find one day that your computer has died you will have a major problem on your hands. Phoning the Sinclair Research customer relations department in Camberley will not help very much either.

If your electronics knowledge is limited, then you will decide to send your Spectrum to one of the companies who specialise in repairing computers. If you have a good working knowledge of electronics then the addresses listed

below should enable you to obtain most replacement parts to repair your computer. Some of the simpler repairs, such as those involving keyboards can be undertaken by those with limited knowledge.

If your computer is under guarantee then it is important that you do not invalidate that guarantee. Talk to the customer service division at

Sinclair Research and they will tell you where to send it. When returning the computer put a note in describing the fault as briefly as possible, as this will speed the repair process.



Suppliers of Sinclair parts:

Adaptors and Eliminators, 14 Thames Street, Louth, Lincolnshire. (050782) 8011. Spectrum and ZX-81 power supplies.

ASTEC (UK), 16 Albany Close, Reading, Berkshire. (0734) 53067. Supplies modulators for all countries PAL, NTSC, SECAM.

CPC, 194-200 North Road, Preston, Lancashire. (0772) 555034. Supplies all parts for Spectrum, ZX-81, ZX printer, ZX RAM pack including all case parts, power supplies and keyboard parts. Probably the best place to get ULA, ROM, case parts and keyboard.

Ferranti Semiconductors, Computer Road, Hollinwood Avenue, Oldham, Lancashire. (061) 682 6844, (061) 624 0515/6661. Supplies Spectrum and ZX-81 ULA.

Maplin Electronics, Southend on Sea, Essex. (0702) 552961. Supplies general components and data and technical books.

National Semiconductors (UK) Ltd, 301 Harpur Centre, Horne Lane, Bedford. (0234) 47147. Supplies Spectrum RAM/74LS series TTL; LM 1889 video chip.

NEC (UK) Ltd, Block 3, Carfin Industrial Estate, Motherwell. (0698) 732221. Supplies Spectrum ROM and RAM/74LS TTL; Z80A.

Sendale Plastics Ltd, Ramsey Road, St Ives, Cambridge. (0480) 62095. Supplies ZX-81 and Spectrum case parts.

Sinclair Research Ltd, Camberley, Surrey. (0276) 685311.

Texas Instruments, Manton Lane, Bedford, Bedfordshire. (0234) 223000, (0234) 211655. Supplies RAM 4116, 4532, 4164, 74LS TTL (Spectrum).

Timex Corporation, Camperdown Plant, Harrison Road, Dundee, Tayside. (0382) 819211. Makers of Spectrum, ZX-81, Printer, Interface one, Microdrives, RAM Pack. Repair and service for Sinclair.

Verran (Computerfix), Units 2H & 2J, Albany Park, Frimley Road, Camberley, Surrey. (0276) 66266. Repairs Spectrums and ZX-81.

Zilog (UK) Ltd, Zilog House, 45-53 Moorbridge Road, Maidenhead, Berkshire. (0628) 39200. Supplies Z80A.

Component companies

SOME OF THE large companies mentioned above do not like dealing directly with the public. The following is a list of component companies which should be happy to deal with you. Most will have catalogues available and will supply data sheets for their products on demand.

These firms are only distributors and they do not make components. If you have a problem with a particular device contact the manufacturer as listed above because not many of the distributors can answer technical questions; RS Components, Farnell, Ambitt and Hawkes may do. Anyone needing NEC parts should phone them direct and ask for some distributors dealing with them.

Abacus Electronics, Kenner House, Pembroke Road, Reading, Berkshire. (0734) 33311. Makes dealt with: National Semiconductors, SGS.

Access Electronic Components Ltd, Austin House, Bridge Street, Hitchin, Hertfordshire. (0462) 57244. Makes dealt with: National Semiconductors.

Alpha Electronic Components Ltd, 66 Wilbury Way, Hitchin, Hertfordshire. (0462) 57244. Makes dealt with: National Semiconductors.

AM Lock Distribution Ltd, Nevill Street, Middleton Road, Oldham,

Lancashire. (061) 652 0431. Makes dealt with: Motorola, Mostek.

Anzac Components Ltd, Burnham Lane, Slough, Buckinghamshire. (06286) 4701. Makes dealt with: Hitachi.

Axion Electronics Ltd, Unit F, Turnpike Road, Cressex Industrial Estate, High Wycombe, Buckinghamshire. (0494) 442181. Makes dealt with: Motorola.

BA Electronics Ltd, Millbrook Road, Yate, Bristol. (0454) 315824. Makes dealt with: Texas Instruments.

Celdis Ltd, 37-39 Loverock Road, Reading, Berkshire. (0734) 585171. Makes dealt with: Motorola, Mostek.

Crellon Electronics Ltd, 380 Bath Road, Slough, Berkshire. (06286) 4434. Makes dealt with: Motorola, SGS, Zilog.

Dialogue Distribution Ltd, Watchmore Road, Camberley, Surrey. (0276) 682001. Makes dealt with: Hitachi.

DTV Group, 10-12 Earnest Avenue, West Norwood, London SE27. (01) 670 6166. Makes dealt with: National Semiconductors.

Farnell Electronic Components Ltd, Canal Road, Leeds. (0532) 636311. Makes dealt with: Hitachi, National Semiconductors.

Hawke Electronics Ltd, Amotex House, 45 Hanworth Road, Sunbury on Thames, Middlesex. (01) 979 7799. Makes dealt with: Motorola, Texas Instruments. Will answer technical queries.

Hill Electronics (NI) Ltd, 290 Antrim Road, Belfast, Northern Ireland. Makes dealt with: Mostek.

ITT Multicomponents, Edinburgh Way, Harlow, Essex. (0279) 442971. Makes dealt with: Motorola, National Semiconductors, SGS, Texas Instruments and Hitachi.

Macro Marketing Ltd, Burnham Lane, Slough, Berkshire. (06286) 4422. Makes dealt with: Motorola, National Semiconductors, Zilog, Texas Instruments.

Quarndon Electronics Ltd, Slack Lane, Derby. (0332) 32651. Makes dealt with: Texas Instruments.

STC Electronics, Edinburgh Way, Harlow, Essex. (0279) 26777. Makes dealt with: Motorola, National Semiconductors, SGS, Texas Instruments and Hitachi.

Computer parts

Z80A: Zilog, SGS, NEC (Spectrum/ZX-81).

ULA: Ferranti (Spectrum/ZX-81).

ROM: Hitachi, NEC (Spectrum), Mostek, Motorola (ZX-81).

RAM: 4116 type — NEC, National Semiconductors, Texas Instruments, ITT (Spectrum), 4532 type — Texas Instruments, OKI (Manhattan Skyline).

ITT (Spectrum), 4532 type — Texas Instruments, OKI (Manhattan Skyline).

2114 type — Motorola, NEC (ZX-81), 4118 type — Mostek (ZX-81), 2K type — Mostek, Toshiba, Motorola, Texas Instruments, NEC (ZX-81).

LM1889: ICI4 — National Semiconductors (Spectrum).

TTL: 74LS00/74LS32/74LS157 — Texas Instruments, SGS, Motorola, National Semiconductors, NEC.

The following are all used on the 48K Spectrum.

Regulator: LM7805 +5V Reg. — SGS, NEC, Texas Instruments, Motorola.

Crystals, Capacitors, Resistors, Diodes, Sockets: available through general component suppliers.

Modulator: UM1233 Astec (UK) Ltd.

Leads: available at most TV/electrical shops.

Case parts: Sendale Plastics.

Keyboard: CPC.

Britain

Aylesbury Computer Club: 12 Long Plough, Aston Clinton, Aylesbury, Bucks.

Aylesbury ZX Computer Club: Ken Knight, 22 Mount Street, Aylesbury (5181 or 630867).

Basildon: Roundacre Microcomputer Users' Club, J Hazell, Basildon 285119/416333. Meetings every Wednesday 7.30 to 10.30pm.

Blackburn Computer Club: 1 Sutton Street, Feniscowles, Blackburn, Lancashire. Tel: B'burn 60033 (office hours) or 28127. Meets twice a month, subscription £5 (£3 juniors).

Bristol Yate and Sodbury Computer Club: 99 Woodchester Yate, Bristol, BS17 4TX.

Cardiff ZX Club: Steve Smith (0222) 593237 or Mike Hayes (0222) 371732. Meets twice a month.

Colchester Sinclair User Group: Richard Lown, 102 Prettygate Road, Colchester CO3 4EE.

Computer Club International, 6 Drumdoon Walk, Downpatrick, N. Ireland BT30 6UF.

Cornard Sinclair User Group: Neil MacDonald, 15 Potkiln Road, Great Cornard, Sudbury, Suffolk CO10 0DA.

Crewe and Nantwich Computer Users' Club: J E A Symondson, 46 London Road, Stapeley, Nantwich, Cheshire CW5 7JL.

Daventry & District Computer Club: c/o Daventry Ex-servicemen's Club, Market Square, Daventry, Northants.

Doncaster and District Micro Club: John Woods, 60 Dundas Road, Wheatley, Doncaster DN2 4DR; (0302) 29357.

Eastwood Town Microcomputer Club: E N Ryan, 15 Queens Square, Eastwood, Nottingham NG16 3BJ.

Edinburgh: Edinburgh Home Computing Club. John Palmer (031 661 3183) or Iain Robertson (031 441 2361).

EZUG-Educational ZX-80-81 Users' Group: Eric Deeson, Highgate School, Birmingham B12 9DS.

Furness Computer Club: R J C Wade, 67 Sands Road, Ulverston, Cumbria (Ulverston 55068). Meets every other Wednesday.

Glasgow ZX-80-81 Users' Club: Ian Watt, 107 Greenwood Road, Clarkston, Glasgow G76 7LW (041 638 1241).

Gloucester: Mid-weekly Spectrum User Group. Barry Ledbury, 8 Linnet Close, Gloucester GL4 9XA (0452) 23186.

Gravesend Computer Club: c/o The Extra Tuition Centre, 39 The Terrace, Gravesend, Kent DA12 2BA. Bi-monthly magazine and membership card.

Hassocks ZX Micro User Club, Sussex: Paul King (Hassocks 4530).

Hobbit Appreciation Society, 12 Middlefield Lane, Hinckley, Leicestershire LE10 0RB. Free newsletter with SAE.

Independent QL Users Group: Brian Pain, 24 Oxford Street, Stony Stratford, Milton Keynes MK11 1JU. Tel: 0908 564271. Publishes newsletter.

Inverclyde ZX-81 Users' Club: Robert Watt, 9 St. John's Road, Gourrock, Renfrewshire PA19 1PL (Gourrock 39967). Meets every other Monday at Greenock Society of the Deaf, Kelly Street, Greenock.

Keighley Computer Club: Colin Price, Redholt, Ingrow, Keighley (603133).

Lambeth Computer Club: Robert Baker, 32 Heatherington Road, London SW4 7NX.

Liverpool ZX Club: Meetings every Wednesday 7pm at Youth Activities Centre. Belmont Road, Liverpool 6. Keith Archer, 031-236 6109 (daytime).

Llanelli Computer Club: 40 Tan-Y-Bryn, Burry Port, Dyfed. Llanelli 56917.

Manchester Sinclair Users' Club: Meets every Wednesday, 7.30pm, at Longsight Library, 519 Stockport Road, Longsight — 061-225 6997 or 061 445 6316.

Meopham: National ZX Spectrum User Club. Guy Fullalove, Woodcotes, Camer Park, Meopham, Kent DA13 0XS. Bi-monthly newsletter, subscription £1.50. Send SAE for details.

Merseyside Co-op ZX Users' Group: Keith Driscoll, 53 Melville Road, Bootle, Merseyside L20 6NE; 051-922 3163.

Micro Users' Group: 316 Kingston Road, Ewell, Surrey KT19 0SY.

Mid-Kent Micro Club: Meets once monthly. Enquiries to M Gates, 65 Buckland Road, Maidstone ME16 0SH.

Mill Lane Association Computer Group: Bryan McAlley, 1 Cow-leaze, Chinnor, Oxfordshire. (0844) 52426.

Newcastle (Staffs) Computer Club: Meetings at Newcastle Youth and Adult Centre, Thursday, 7.30. Enquiries to R G Martin (0782 62065).

North Hertfordshire Home Computer Club: R Crutchfield, 2 Durham Road, Stevenage; Meetings: first Friday of the month at the Settlement, Nevells Road, Letchworth.

Northern Ireland Sinclair Users' Club: P Gibson, 11 Fitzjames Park, Newtownards, Co Down BT23 4BU.

North London Hobby Computer Club: ZX users' group meets at North London Polytechnic, Holloway Road, London N7 Monday, 6pm.

Nottingham Microcomputer Club: ZX-80-81 users' group, G E Basford, 9 Holme Close, The Pastures, Woodborough, Nottingham.

Orpington Computer Club: Roger Pyatt, 23 Arundel Drive, Orpington, Kent (Orpington 20281).

Perth and District Amateur Computer Society: Alastair MacPherson, 154 Oakbank Road, Perth PH1 1HA (29633). Meetings: third Tuesday of each month at Hunters Lodge Motel, Bankfoot.

Regis Amateur Microcomputer Society: R H Wallis, 22 Mallard Crescent, Pagham, Bognor Regis, West Sussex PO21 4UU.

Roche Computer Club: 8 Victoria Road, Coop Rooms, Roche, Cornwall: 0726 890473. Twice weekly meetings, Monday and Friday.

Saltcoats Computer Club: Colin Borland, 117 High Road, Saltcoats, Ayrshire KA21 5SD. Weekly meetings.

Sinclair Postal User Group: 24 St. Mary's Way, Code SUL, Chigwell, Essex IG7 5BX. Produces magazine with competitions.

Scunthorpe ZX Club: C P Hazleton, 26 Rilestone Place, Bottesford, Scunthorpe; (0724 63466).

Sheffield: South Yorkshire Personal Computing Group. R Alderton (0742 20571), S Gray (0742 351440), P Sanderson (0742 351895).

Sinclair Amateur Radio User Group: SAE or two IRCs for details. Paul Newman G4 INP, 3 Red House Lane, Leiston, Suffolk IP16 4JZ.

Sittingbourne: Anurag Vidyarth (0795 73149).

St Albans: Bi-monthly meetings and a magazine. Details from Adam Slater, 40 Watford Road, St Albans, Herts AL1 2HA. (0727 54176).

Stratford-on-Avon Computer Club: Meets on the second Wednesday of every month. Telephone: 0789 68080 for details.

Swansea Computer Club: B J Candy, Jr Gorlau, Killay, Swansea (203811).

Swindon ZX Computer Club: Andrew Bartlett, 47 Grosvenor Road, Swindon, Wilts SN1 4LT; (0793) 3077. Monthly meetings and library.

Sutton: Sutton Library Computer Club, D Wilkins, 22 Chestnut Court, Mulgrave Road, Sutton, Surrey SM2 6LR.

Washington Sinclair Users' Club, Columbia Community Centre, Tyne and Wear. Meets twice a month, tel. 4179483 or 4167367.

West Sussex: Midhurst and District Computer User Group. Enquiries to V Weston (073 081 3876), R Armes (073 081 3279).

Worle Computer Club: S W Rabone, 18 Castle Road, Worle, Weston-super-Mare BS22 9JW (Weston-super-Mare 513068).

Universal ZX Club: Postal club for Spectrum owners in the U.K. and abroad. C. Shaw, 1 Swiss Walk, Batley, W. Yorkshire.

ZX-Aid: Conrad Roe, 25 Cherry Tree Avenue, Walsall WS5 4LH. Please include sae. Meetings twice monthly.

Overseas

Australia: Australian ZX Users' Newsletter, incorporating QL User. Paul Janson, P.O. Box 397, Dapto 3530, Australia. Also seeks unpaid contributions for the newsletter.—W.A. ZX Users' Group, Garth Gregson, 34 Chester Street, South Fremantle 6162. Phone 3351671.

Austria: ZX User Club, Thomas Christian, c/o Wissenschaft Forscht e. V., Postfach 141, A1190 Vienna. Meets every first Friday of the month. Telephone 0222-44 32 050 for details.

Belgium, France and Luxembourg: Club Micro-Europe, Raymond Betz, 38 Chemin du Moulin 38, B-1328 Ohain, Belgium (32/2/6537468).

Denmark: Danmarks National ZX-8081 Klub (DNZK), Jens Larson, Skovmosevej 6.4200 Slagelese, post giro 1 46 24 66.

ZZ-Brugergruppen i Danmark, Boks 44, 2650 Hvidovre, Gratis medlemskab og gratis blad til enhver interesseret.

J Niels-Erik Hartmann, OZ-ZX-Radioamator, Bruger Gruppe, Bredgade 25 DK-4900, Nakskov.

Finland: ZX-kerho, c/o Kalevi Hamalainen, Siltakatu 9 A 8, 33100 Tampere 10, Finland. Phone 35831-34238. Publishes quarterly paper.

France: Yves Chapron, no. SUS-1047, Rue du Puy, La Terrasse, 38660 Le Touvet, France. Specifically for users in the Alps.

Germany: ZX Club, a postal club; contact Aribert Deckers, Postfach 967, D-7000 Stuttgart 1, West Germany.

Greece: Athens Spectrum Club, Paris Stamelos, Spetsou 2, isi22 Marousi, Athens, Greece.

Indonesia: Jakarta ZX-80/81 Users' Club, J S Wijaya, PO Box 20, Jkukg, Jakarta, Utara, Indonesia.

Irish Amateur Computer Club: Martin Stapleton, 48 Seacourt, Clontarf, Dublin 3. (331304).

Irish Sinclair Users Club: PO Box 1238, Dublin 1. Publishes a newsletter. Send SAE for details.

Italy: Sinclair Club, Via Molino Vecchio 10/F, 40026 Imola, Italy.

Genova Sinclair Club: Vittorio Gioia, Via F Corridoni, 2-1, telephone 010 3125 51.

continued on page 138

Club Corner

continued from page 137

Micro-Europe: Belgium or Club Paris-Micro, 19 Rue de Tilly, 92700, Colombes, France; associated with Club Micro-Europe.

The Netherlands: Clive's Bits and Bytes, Nicholas Beets Straat 164, 6416 VX Heerlen. Telephone 045-423024. Both Spectrum and ZX-81 users welcome.

Republic of Ireland: Irish ZX-8081 Users' Club, 73 Cnoc Crionain, Baile Atha, Cliath 1.

Singapore: Sinclair Users' Group: Eric Mortimer, 1D Wilmer Court, Leonie Hill Road, Singapore.

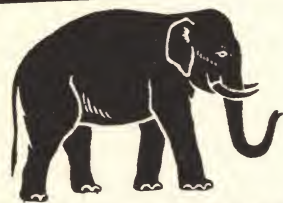
South Africa: Johannesburg ZX-80-81 Computer Users' Club: S Lucas, c/o Hoechst SA (Pty) Ltd, PO Box 8692, Johannesburg.
Johannesburg ZX Users' Club: Lennert E R Fisher, PO Box 61446, Marshallstown, Johannesburg.
Dumont and Syndercombe Amateur Computer Club: Jean-Pierre Dumont would like to correspond with ZX-81 owners via tapes. Write to 8 Kipling Road, Farrarmere, Benoni 1500, Transvaal.

ZX SA Club: Jonathan Jones, House 14, Anglo Alpha PO Box 15, 1725 Roodepoort.

Spain: Club Nacional de Usuarios del ZX-81, Joseph-Oriol Tomas, Avda. de Madrid, No 203 207, 10, 3a esc. A Barcelona-14 Espana.
International ZX Spectrum Club: Gabriel Indalecio Cano, Sardana, 4 atrico 2a, San Andres de la Barca, Barcelona. Send international reply coupon. Produces a bi-monthly magazine. Spanish ZX Micro Club: Apartado 181, Alicante (Costa Blanca), Spain.
ZX Club Spain; C Benito PO Box 3253, Madrid, Spain.

Swedish ZX-club: Sinclair Datorklubben, Box 1007, S-122, 22 Enskede.

United States: Bay Area ZX-80 User Group, 2660 Las Aromas, Oakland CA94611.—Harvard Group, Bolton Road, Harvard MA 01451: (617 456 3967).—SAF Users' Group, 2749 Eden Road, Leslie, Michigan 49251.—ZX Users Group of New York, Box 560 Wall Street, New York, N.Y. USA 10005. Subscription \$15US, publishes international newsletter. Seeks newsletter exchange with other groups.



ELEPHANT SOFTWARE

FOR THE SPECTRUM 48K
WRITTEN BY PROFESSIONALS

**TWO SOFTWARE PRODUCTS FOR
THE SERIOUS USER AT AN
AMAZINGLY LOW PRICE!!**

FINANCE MANAGER

For Small Business and Personal Use.

- Easy to use • Cash and Bank Balance Report
- List of Unpaid bills • Simple Trading A/C & Profit & Loss A/C Production • Graphic Business Performance Analysis

FILE MANAGER

1,001 uses in the home or business

- Easy to use • Includes all standard facilities
- Sort and search capability • Stores up to 1,300 items

BOTH PROGRAMS FOR AN INCREDIBLE £6.95

Trade Enquiries Welcome

*All tapes
Guaranteed*

To: Elephant Software, Ashlea, Japps Lane, Barton, Preston, Lancs.
Please supply FINANCE MANAGER and FILE MANAGER.
I enclose cheque for £7.40 (incl. 45p post & pack).

NAME.....
ADDRESS.....
POSTCODE.....
SIGNATURE.....

Advertisement Index

Adder Publications	28
Advanced Memory Systems	4
Campbell Systems	29
Cheetah Marketing	43, 86
Compusound	29, 135
Cross Software	28
Currah	66
DK Tronics	IFC
Eprom Services	25
Elephant Software	138
Firebird Software	58, 59
Fox Electronics	21
Gamma Software	57
Incentive Software	91
Metacomco	38
Modular Resources	60
National Software Library	138
Oxford Computer Publishing	92, 101
Pitch Association	30
Practical Software	91
Quest	IBC
Quick Silva	OBC
Ram Electronics	12
Rainbow Electronics	25
Saga Systems	17
Selec Software	25
Sigma Research	29
Software Farm	52
Spectadraw	91
Software Library	135
Tasman Software	102

HIRE SPECTRUM SOFTWARE

OVER 280 different tapes for hire including **ARCADE, ADVENTURE, BUSINESS, EDUCATIONAL**, etc.—even an m/c Compiler. **FREE** 26 page catalogue. **FREE** newsletter, with hints, tips and our top tape chart.

OVER 3000 tapes in stock, with up to 60 copies of each title for fast service. All tapes sent by 1st class post.

LOWEST PRICES—hire up to 3 tapes at a time, from 63p each for two weeks hire. (Plus p&p and VAT). European members welcome.

Tapes for sale at DISCOUNT prices. Telephone 01-661 9240 (9am - 5pm) or write for further details, or complete the coupon and **JOIN TODAY**—you've nothing to lose!

SAVE £3.00!!

For a limited period, we are offering **HALF-PRICE** membership to readers of this magazine. Join now, **LIFE** membership is only £3.00 (normally £6.00).

SWOP SHOP

A new service from N.S.L. **SWOP** your unwanted tapes with tapes from other members for a small handling charge of only 60p (plus p&p and VAT)

NATIONAL SOFTWARE LIBRARY

42 Harefield Avenue, Cheam, Surrey SM2 7NE.

I enclose cheque/postal order for £3.00 for LIFE membership. Please rush my membership kit to me. If, within 28 days, I'm not delighted with your service you'll refund my membership fee.

Name.....
Address.....

SU 12

QUEST ANNOUNCE 6 WAYS OF GETTING THE MOST FROM YOUR SINCLAIR QL

CP/M Operating System

Business Software

Disk Storage

Mass Storage

Expansion Unit

Memory Add-ons

**Already Available—all essentials for
your QL range...** ■ Sinclair QL's
■ Printers ■ Monitors ■ Cables, paper etc

For priority information pack write, telephone
or see us at COMPEC stand 938

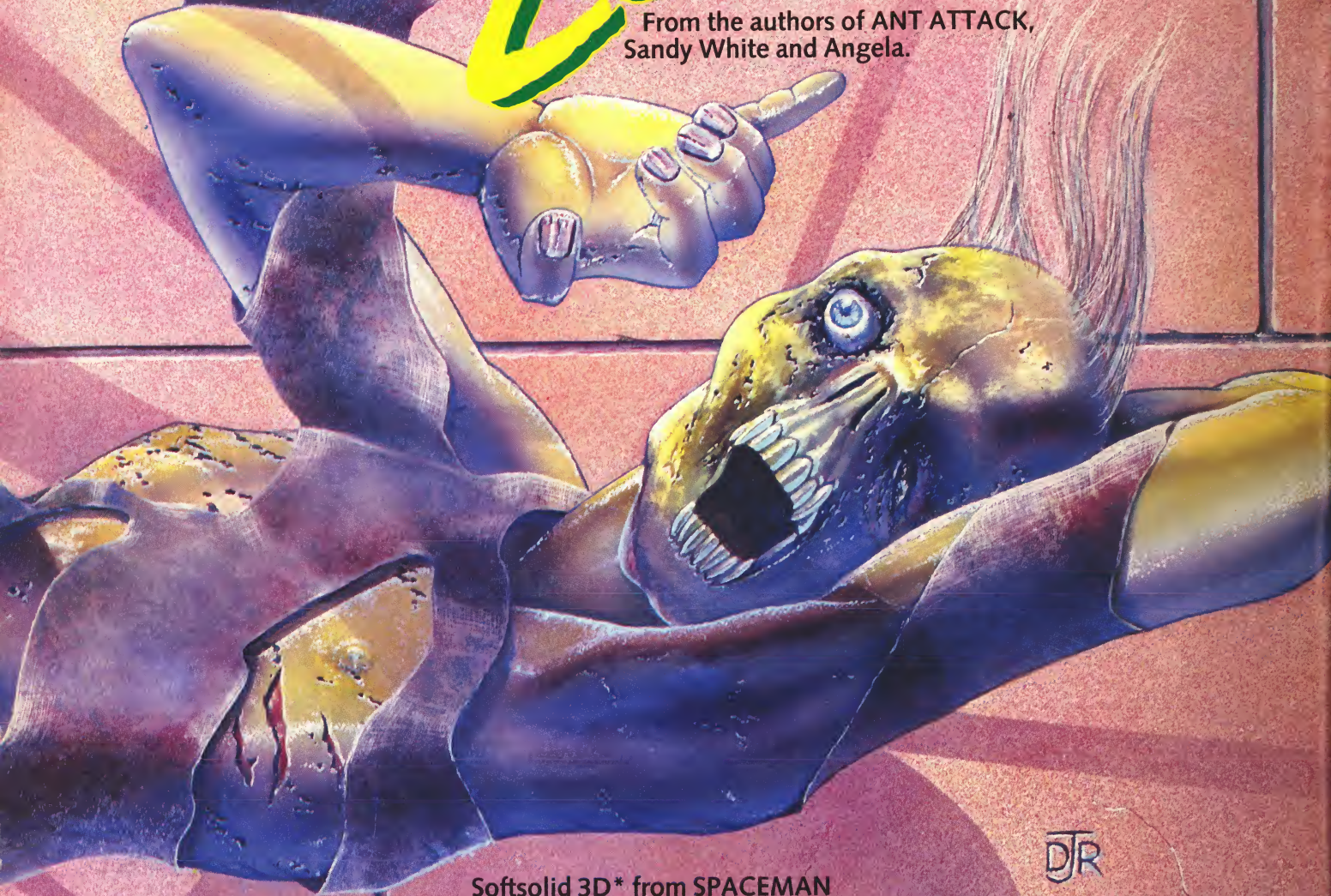
Quest International Computers Ltd.
Quest House, School Lane, Chandlers Ford, Hants SO5 3YY
Tel: Chandlers Ford (04215) 66321 Telex: 47326

The Computer Professionals
quest

A prophecy is about to be fulfilled. The Dead will rise again to eat the flesh of the living...

ZOMBIE ZOMBIE

From the authors of ANT ATTACK,
Sandy White and Angela.



Softsolid 3D* from SPACEMAN
*Patent pending

48K Spectrum **£6.95**

All titles available from

QUICKSILVA

Quicksilva Mail Order, P.O. Box 6, Wimborne, Dorset BA21 7PY.
Telephone (0202) 891744.

WARNING: These programs are sold according to QUICKSILVA Ltd's terms of trade and conditions of sale, copies of which are available on request.

WHSMITH



WOOLWORTH

John Menzies

and leading multiple retailers and
specialist computer stores.

An EMAP
Publication

£2.25

The 1985 sinclair user annual

ALL THAT MADE THE NEWS IN THE SINCLAIR YEAR

25 PAGES OF NEW
PROGRAM LISTINGS

THE FULL
QL STORY

BUYER'S GUIDE
TO JOYSTICKS
PRINTERS AND
KEYBOARDS

TOP SOFTWARE
AUTHOR
INTERVIEWED

HITS AND MISSES
IN SOFTWARE IN 1984



ALL YOU NEED TO KNOW
ABOUT SINCLAIR
COMPUTING